1995

A two stage approach for improving hospital inpatient operations efficiency

Wen-Yen Chen
Lehigh University

AUTHOR:

Chen, Wen-Yen

TITLE:

A Two Stage Approach for Improving Hospital Inpatient Operations Efficiency: Capacity Planning and Concurrent Scheduling

DATE: May 28, 1995

A Two Stage Approach for Improving Hospital Inpatient Operations Efficiency : Capacity Planning and Concurrent Scheduling

By

Wen-Yen Chen

A Thesis

Presented to the Graduate and Research Committee

of Lehigh University

in Candidacy for the Degree of

Master of Science

in

Program of Manufacturing Systems Engineering

Lehigh University

January, 25th, 1994

This thesis is accepted and approved in partial fulfillment of the requirements for the Master of Science.

Jan. 26, 1995

Date

Thesis Advisor

Director of Program

Chairperson of Department

Acknowledgements

There have been many people who have helped me to go through two years of foreign student life. Without their support, encouragement, and guidance, I doubt that I can finish this study. My parents, Wan-Jiu and Shou-Ying Chen, always support me. My best friends, Tom Lee and Po-Jang Huang, are the most generous people I ever seen. Finally, I wish to thank my advisor, Dr. Wilson. He always is so patient and ready to help me, although I given him some hard times. There are so many people who help me through these two years- some I know and some I don't. Thus, what I have to say is thanks to God.

Table of Contents

List Of Tables

List Of Figures

Abstract

This paper provides a model for hospital administration to do capacity planning, resource allocation and patient scheduling. The major figures of this model are Semi-Markov model for demand forecasting and Concurrent Scheduling for patient scheduling. The model avoids redundant capacity, facilitates inpatient flow, and reduces waiting time.

A Semi-Markov model provides monthly total demands for each kind of resource. When demands are determined, a linear program uses this data to determine the capacity for each kind of resource.

To evaluate the patient scheduling portion of this model, three other heuristic models, the Weighted First Come First Serve (WFCFS), Weighted Shortest Process Time (WSPT), and the mixed method of WSPT and Balanced Work Load (BWL), are created and tested. By using the SIMAN-4 simulation package, the performance of these models are evaluated. Experiments show that the overall model which includes a Semi-Markov model and a linear programming model for Capacity Planning and a heuristic scheduling model for Concurrent Scheduling outperforms other models in patient waiting time and costs, especially when patient queues begin to pile up.

A Two Stage Approach for Improving Hospital Inpatient Operations Efficiency : Capacity Planning and Concurrent Scheduling

1. Introduction:

In the US and with regard to general public expenditures, the health care industry ranks third following education and the national defense industry. In 1990, Americans spent $ 666 billion on health care. Unfortunately, health care costs continue to soar.

The two major parts of government health care expenditures are the following:

1) Medicare: the federal social security health insurance plan for elderly, disabled, and other groups, and

2) Medicaid: the federal / state welfare program for health care.

In 1981, to control soaring heath expenditures, the US government analyzed the costs of each Disease Related Group (DRG) and set a fixed payment for each DRG. DRG assignment variables include:

1) medical or surgical procedure,

2) age,

3) comorbidity, and

4) complication.

Using the above variables, a patient's DRG can be determined. In 1984, the government began to enact this method to reimburse hospitals. By 1987, most of the states already were using DRG method to pay hospitals for these two kinds of expenditures. But Williams, et al. (1993) noted that the spending for Medicare and Medicaid has been increasing at an even more rapid rate than that for total national health expenditures. So, the US government has attempted to freeze DRG payments to control the spending.

Private health insurance corporations, which provided 33% of total health care expenditures in 1990, used a similar method, the Perspective Payment System (PPS), to pay hospital costs. This method pays hospitals a fixed payment per admission.

To survive in this environment, hospitals have to pay more attention to cost management. The administrative staffs of hospitals now spend more time in controlling costs than in the past. They stress hospital efficiency, bed occupancy ratio, and patients' Length Of Stay (LOS).

Some of them use Total Quality Management (TQM) to keep old customers coming back and attract new customers in order to increase their hospital bed occupancy ratios. This is true especially in some suburban and rural hospitals, because of their low bed occupancy ratios. Total Quality Management stresses the satisfaction of customers. This is a good direction for hospital management. Unfortunately, there is a disadvantage to this approach. Most metropolitan hospitals are too busy to use this method; on the other hand, some suburban and rural hospitals, whose capacity is far more than their needs, prefer it very much. Because redundant resources cost too much for these suburban and rural hospitals, some are going out of business. Thus, there is a strong justification for a good capacity plan.

Some hospitals try to take advantage of PPS by shortening patient's LOS. This payment system pays hospitals a fixed reimbursement per admission for patients in a particular DRG. So, the longer a patient stay, the more costly the patient becomes. But if patient days (or LOS) are cut too much, it may cause higher recurrence rates. To administration, a little higher recurrence rate doesn't necessarily mean troubles, because a higher recurrence rate means more admissions; the higher the recurrence rate is, the more reimbursements hospitals can receive. Of course, when recurrence rates become too high, they will reduce health care service quality and hurt relationships between patients and hospitals.

Recently there is a new approach, which will keep advantages of the above methods and avoid disadvantages of these methods. Some hospitals began to unify into hospital groups. These hospitals account for 48% of the United States' community hospitals in 1992. In a hospital group, each hospital has its own areas of focus, such as a cancer hospital. There are many advantages to this approach.

1) This increases facility utilization and total patient volume of each DRG.

2) When a hospital group decides to purchase new equipment which will attract more customers, the risks of the investment can be reduced by sharing these risks throughout a group, not just at a single hospital.

In addition, some insurance corporations have begun to contract with hospital groups. They pay a hospital group a fixed annual premium and the hospital group will provide health care for the entire year. This is a brand new market for hospitals.

Objective of my thesis is to provide hospital administrative staffs a simple and useful method to resource utilization, schedule patient list, and facilitate patient flow.

This method is a two stage approach which includes Capacity Planning and Patient Scheduling.

When patient arrival frequencies change (demands change), there is feedback signal sent to update capacity planning.

The major procedure of my model is given on figure 1.

The first stage of the method is "capacity planning", which will avoid redundant resources and meet the needs of patients. What follows is a procedure for capacity planning:

1) Data collection

a) the first moment (or average) of the process time of each resource in a treatment unit. The formula of the first moment (or average) process time is:

$$t_{ij} = \sum_{k=1}^{n} \frac{t_{ijk}}{n}$$

t_{ij} = the first moment of the process time that a particular DRG type patient stays in unit i, before he is transferred to another treatment unit or an absorption state j; unit(min)

t_{ijk} = the process time that a particular DRG type patient k stays in unit i, before he is transferred to

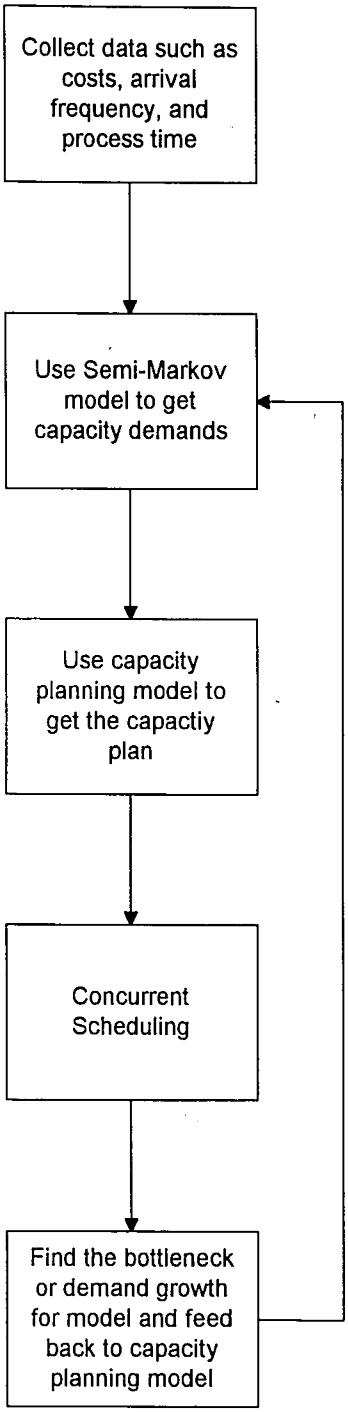

Figure 1 : the procedure of the two stage model - Capacity Planning and Concurrent Scheduling

another treatment unit or an absorption state j; unit(min)

n = the monthly total number of a particular DRG type patient

b) the second moment of the process time of each resource in a treatment unit. The formula of the second moment of process time is:

$$t_{ij}^{2} = \sum_{k=1}^{n} \frac{t_{ijk}^{2}}{n}$$

t^2_{ij} = the second moment of the process time that a particular DRG type patient stays in unit i, before he is transferred to another treatment unit or an absorbing (discharge) state j; unit(min^2)

t^2_{ijk} = the square of the process time that a particular DRG type patient k stays in unit i, before he is transferred to another treatment unit or an absorption state j; unit(min^2)

c) the transition probability of each DRG. It is the probability that a particular DRG type patient, who finishes his treatment in treatment unit i, will transfer to another treatment unit or absorbing (discharge) state j.

d) the probability of a patient associated with a particular DRG enters a hospital is treatment unit i. It is the probability that the treatment unit i is the first treatment unit where a particular DRG type patient enters a hospital.

e) the monthly average arrival frequency of each DRG patient type; unit(patient/month)

2) Monthly resource requirement forecasting

a) By using (1.a), (1.b), and (1.c) in a Semi-Markov model, the expected facility usage time for each DRG patient can be determined.

b) With the monthly expected facility usage time for each DRG patient and the monthly patient arrival frequency of each DRG, the monthly demands for resources can be obtained.

c) Finally, the capacity plan of inpatient operations is determined by a linear programming.

The second stage of this method, called "Concurrent Scheduling", will facilitate patient flow and decrease costly waiting time. Some treatments simultaneously require multiple resources. For example, a surgery needs a surgeon, an anesthetist, several special nurses, equipment, and an operating room at the same time. Traditional scheduling methods can't solve this problem effectively, because of the complexity of the problem.

The Concurrent Scheduling model uses three strategies to handle the multiple resource problem.

a) Gather required resources into resource group by the rule "See resources, which are always ready at the same time and location, as a single resource".

b) Balance the work load of same type of resources.

c) Use a priority rule for Concurrent Scheduling to allocate required resources to the highest priority patient in patient queues.

To verify the superiority of the Concurrent Scheduling model, three other scheduling models are created:

a) The Weighted First Come First Serve (WFCFS) method,

b) The Weighted Shortest Process Time (WSPT) method, and

c) The mixed method of WSPT method and the Balanced Work Load (BWL) method.

WFCFS is the most commonly used scheduling method for inpatient operations. According to Morton and Pentico (1993), WSPT has shown that it is a robust scheduling method in an industrial setting. Therefore, WSPT and the mixed method of WSPT and BWL are used to compare to the performance of Concurrent Scheduling.

It will be shown through simulation experiments that Concurrent Scheduling tends to give better results than the rest of the methods.

§2 reviews research related to patient flow, cost management, and job scheduling. There are some discussion about inpatient operations in §3. In §4, there is a detailed description of the proposed capacity planning and concurrent scheduling model. In §5, a simple inpatient operations example is defined and analyzed. To estimate the performance of this model, three heuristic scheduling models were created. Results of simulations compare the total waiting costs, which are the only variable costs in the inpatient operations, for the four models. In the final section, the results of the experiment are discussed and future directions for inpatient operations are explored.

2. Related Work:

Capacity Planning plays an important role in hospital cost management. Without correct patient demand prediction, capacity planning will lead to costly redundant capacity or capacity shortages. For example, capacity shortages will cause customers waiting and customer loss to another hospital. Good capacity planing begins with an accurate prediction of patient flow and routes. There is much research about how to approximate the patient flow and routes. It can be divided into two categories - simulation methods and analytic methods.

Simulation methods use empirical data to emulate correlations across time, patients, and resources. This is the most economic and practical means to observe these correlations. Another important function of simulation methods is the ability to verify capacity plans in an inexpensive way. In my thesis, a simulation method will be used as a validating tool.

Analytic methods as applied to hospital operations can be separated into three groups - Queuing theory, the Markov model, and the Semi-Markov model.

Queuing theory uses patient arrival distributions, patient process time distributions, number of identical resources, restrictions on the sizes of queues, and priority rules to represent practical conditions. It assumes that a system will reach a steady-state condition after a length of time, for instance one quarter of a year. It will provide the expected number of customers in queues, the expected number of customers in services, and the expected waiting time in service queues. In addition, Queueing theory can merge many treatment units into a specific patient path. But when patient paths become complex, Queueing theory will be more complex than the corresponding Semi-Markov model.

The Semi-Markov model, like the Markov model, uses a transition probability matrix to decide an entity's (or patient's) next state or treatment unit (such as a surgery unit, an internal medical unit, and an intensive care unit (ICU)). It is assumed that the probability that a patient transfers from treatment unit i to treatment unit j depends only on the current treatment unit, no matter how the patient enters this unit. The major difference between the Semi-Markov model and the Markov model is that patient flow in the Markov model changes every time interval, but the patient flow in the Semi-Markov model changes according to patient's stay time in the current treatment unit, not a fixed time interval. The Semi-Markov model takes every patient's sojourn time into consideration, but the Markov model doesn't. So, the Semi-Markov model provides a closer approximation of patient flow for inpatient operations.

Kao, E.P.C. (1973) and (1974), successfully used the Semi-Markov model to approximate the movement of coronary patients within a hospital. It provided a good model to approximate average LOS for coronary patients. In Kao's model, he only considered one resource, patient beds. In addition, he assumed that every patient always can get required resources immediately. Fortunately, this assumption is relaxed by considering Cohen (1991) who provides a tool to consider the relation between capacity and waiting time. In my simplified inpatient operations example, some resource capacities are only one, therefore Cohen's method would not be useful.

Hershey, et al. (1981) expanded Kao's model to a network model, not just a coronary patient flow. They also added a condition into Kao's model that one and only one treatment unit could be short of capacity. So some patients, who needed this treatment, had to wait in the patient queue. In the real world this condition was not very practical, because realistically two or more units could have capacity limits.

Polesel and Romanin-Jacur (1986) used Kao's model and showed that the Semi-Markov model is a pretty promising method to approximate the flow of the dialysis-transplantation patient group.

Howard (1971) is a good general reference for the continuous time Semi-Markov model.

Another important factor in capacity planning is budget allocation. Following are some articles about hospital financially-related issues.

Morey and Dittman (1984) proposed a linear programming price policy for hospital administrative staffs. This policy could avoid losing cash when PPS reimbursements were lower than real costs of therapy.

This model divided patients into two groups according to their payment resources:

1) patients who didn't pay hospitals directly: these charges were paid by Medicare, Medicaid and private insurance corporations.

2) patients who paid hospitals themselves.

At the first step of this method, average costs for each DRG are calculated. At the second step of this method, the difference between real costs and PPS reimbursement for each DRG is determined. If this difference was positive, group 2 patients, who paid hospitals by themselves, were charged more than they should be charged to make up the above cost deficiency. Because this model was unfair to group 2 patients, it is now banned to use this method.

Kao and Queyranne (1985) proposed several decision support models for budgeting the nursing work force requirement. In nursing care services, there are three kinds of nurses: registered nurses, licensed nurses, and nurse assistants. What follows are characteristics of these models:

1) single or multiple period: if the fluctuation of patient arrival frequency was not obvious, then this characteristic was single; else, this characteristic was multiple.

2) aggregate or disaggregate: "aggregate" means the demands for nursing care services are expressed over the nurse skill class mix, such as 100 nurse hours; and "disaggregate" means the demands for nursing care services are expressed by each nurse skill class, such as 10 registered nurse hours, 30 licensed nurse hours, and 60 nurse assistant hours.

3) Deterministic or Probabilistic: "Deterministic" means the arrival time and process time of patients are known or fixed; "Probabilistic" means these times are random variables.

One of these models was called the Multiple, Disaggregate, and Probabilistic (MDP) model. The characteristics of inpatient operations are similar to those of the MDP model. For example, the patient arrival frequency can have seasonal surges; the demands on inpatient operations include every kind of required resource and these demands can't be aggregated together; the process time of treatments and patient arrival intervals are probabilistic, not deterministic. So, this model can provide a robust capacity planning of inpatient operations.

Woodbury and Manton (1992) provided a linear programming model to resolve budgeting allocating problems of military hospitals. In military hospitals, some of the resources are idle and wait for wars or some sudden emergency, such as airplane crash accidents. Because of this redundant capacity, these hospitals need closer cost controls than regular hospitals to avoid losing money.

Charnes, et al.(1978) proposed the Data Envelopment Analysis (DEA) method to evaluate the efficiency of a decision making unit. Banker, et al.(1984) extended DEA to estimate the correlations of cost and production. The Data Envelopment Analysis method is similar to a benchmark method which will measure the performance of a unit by comparing it to the best one. What follows are the steps of DEA:

1) Define inputs, such as labor, and outputs, such as reimbursements.

2) Estimate inputs and outputs by the same unit, such as \$100 per man hour and \$5000 for a surgery.

3) Use following formula to calculate each hospital's efficiency.

Efficiency = {Sum of Hospital Outputs} / {Sum of Hospital Inputs}

4) Compare efficiency across hospitals.

Compared with traditional econometric methods, which have to define many parameters and explore the correlations between parameters, DEA is simple to use for administrative staffs of hospitals.

Besides capacity planning, scheduling also plays an important role in hospital cost management. Although there are many articles which provide promising solutions for financial planning, there is not much research done in the inpatient scheduling area. Since there are not many articles in this area, analogous areas, such as manufacturing industry, were searched to find the most appropriate model of inpatient scheduling.

In the real world, inpatient operations is similar to the Flexible Manufacturing Systems (FMS). The fundamental definition of FMS by Buzacott and Shanthikumar (1980) is "a set of machines ... linked by a material handling system and all under center computer control". Some similar characteristics of these two systems follow:

1) Some operations need multiple resources simultaneously. In a FMS, a working process will need one or more parts, a fixture, a machine, and one or more tools. In inpatient operations, a diagnostic tests, such as chemical and bacteria inspection, needs equipment, operators, and reagents.

2) Some resources can execute multiple functions, such as a CNC machine center in FMS and physicians for inpatient operations.

3) Some resources can be used by many kinds of working processes or treatments. For example, operating rooms can provide services to different kinds of DGR patients. Tools can be used by different kinds of machines.

What follows are some relevant articles for FMS scheduling.

Hutchison, et al.(1989) proposed a linear programming model which reached a near optimal schedule for a static job scheduling problem. Because the probabilistic characteristic of patient arrival distribution, this model can't totally satisfy the requirements of inpatient operations.

Tenenbaum and Seidman (1989) created a dynamic load control policy which dealt with a two step manufacturing system. The procedure for this system is the following:

1) Several traditional dedicated machines processed

raw materials to common parts for step (2).

2) Several flexible manufacturing CNC machine centers processed common parts to different varieties of final products. Each of these machine centers was set up for a specific operation. There is a buffer for common parts in each CNC machine center.

The capacity of specific CNC machine centers, buffers, and traditional machines would lead to several operational scenarios. For example, when all buffers are full, all traditional machines must stop production, because there is no room for any incoming part. After Tenenbaum and Seidman explored several of these situations, a mathematic model was developed. But when the number of operations becomes larger, such as 10-step production operations, this model will become too complicated and beyond control.

Mukhopadhyay, et al.(1991) proposed a simple but useful heuristic method to help jobs find the most appropriate resource among similar function resources, such as CNC machine centers. To integrate all resources, such as machines, pallets, and tools, into a system, a system level strategy is employed which uses the above priority rules to choose machines and tools and schedules accessory resources, such as Automatic Guided Vehicles (AGVs), to facilitate parts moving between machine centers.

Stecke and Solberg (1981) and Shanker (1985) each affirmed the importance of balancing work load of the same resource type. They found that when work loads were balanced, classic dispatch heuristic methods, such as Shortest Process Time (SPT), made not as much of a difference in total flow time.

Besides these traditional scheduling methods, there has been a new method appearing in the literature more recently for FMS, namely Concurrent Scheduling.

Mirchandani and Lee (1988) provided a new scheduling heuristic method, the Concurrent Scheduling heuristic method, for a two-machine Flexible Manufacturing Cell(FMC). Why Concurrent Scheduling? When the functionality of a FMS is fully utilized, it can employ many tools to execute its working assignment. The relationship between tools and machines is, thus, much more complicated. For example, a CNC machine center can hold up to 100 tools in its tool magazine and tool capacity will affect productivity greatly. So it will be too complicated and not efficient to use traditional scheduling methods which see job scheduling and resource allocation as two different steps. It is true that when job scheduling is optimal on a primary resource sometimes the rest of resources can not cope with the job schedule. With the ability to integrate job scheduling and resource allocation, the Concurrent Scheduling method is a very appropriate tool for FMS scheduling problems.

Stecke (1983) identified five inter-related production problems which must be solved prior to system operations:

1) part type selection,

2) machine grouping,

3) production ratio determination,

4) resource allocation, and

5) loading.

Concurrent Scheduling methods have the ability to solve item (1) and item (4) simultaneously- item (1) is part scheduling and item (4) is resource allocation. These methods will simplify these problems for FMS.

In Mirchandani's article, the major strategy of the Concurrent Scheduling method was to balance the work load of parallel multi-function machine centers. Unfortunately, it was a static model and limited to a two-machine problem. But this article again affirmed an important production planning concept, balancing the work load of the same type of resources.

Since the Concurrent Scheduling method can simplify the complexity of FMS, it should also simplify the complexity of inpatient operations.

3. The Inpatient Operations Description:

Although inpatient operations is similar to FMS, inpatient operations are more complicated than FMS in some areas, such as the specific relations between inpatients and physicians. For example, the same category of parts can be processed by different flexible manufacturing machines, but inpatients won't usually change physicians during the hospital stay.

Following are the main characteristics of the inpatient operations assumed in the model to be discussed:

1) Each patient has his specific route through the hospital, which is decided by transition probability.

2) When a patient finishes a treatment in a treatment unit, such as a surgery unit, he will be transferred to another unit or be discharged. He can't immediately reenter the same unit. The transition probability depends only on the patient's DRG and the unit where the patient currently resides.

3) Some treatments simultaneously need multiple resources to execute. The treatment can't be executed until all required resources are ready.

4) Some medical staff members can perform many jobs. For example, nurses can take care of patients, administer medication, and check patients' physical conditions.

5) The major costs of inpatients include:

a) direct nursing costs,

b) indirect nursing costs,

c) ancillary costs,

d) room and beds costs,

e) fixed overhead costs, and

f) variable overhead costs.

6) There are two kinds of admission. One is emergency admission and the other is regular admission. Emergency admission patients need treatment immediately, but regular admission patients can wait longer.

7) On the weekends, a small staff of physicians must stay in the hospital in case of emergency.

8) Basically there is no "due day" for inpatients. But regular medical costs and variable overhead costs accumulate by days, not by cases. In addition, some severe diseases will cause patients to die or make their physical condition worse, if patients don't receive appropriate treatment in a timely fashion. So, hospitals will try to

shorten waiting time as much as possible.

9) Nurses work 3 shift per day. So there always are nurses taking care of every inpatient. To provide a variety of work experiences, nurses have to transfer to another treatment unit every so often, such as every two months. But during a particular time interval, the nurse cannot work in another unit except in an emergency.

10) Physicians and equipment operators work a regular shift per day.

11) Whenever a patient chooses or is assigned to a physician, he can not change his physician while in the hospital.

12) Some equipment used in inpatient operations is very expensive. Hospitals would like to increase this equipment utilization as much as possible.

13) The thing that inpatient operations provides to customers is a service, not a solid product; so, customer satisfaction becomes an important consideration of inpatient operations.

14) Physicians would transfer to another unit every so

often, say every two months, to strengthen work experience variety. But during a particular time interval, he can't work in another unit except in an emergency.

4. The Mathematical Model:

4.1 Capacity Planning:

Inpatient operations is a very complicated and versatile system. Capacity planning is the first step to increase resource efficiency and begins with correct patient flow prediction. The Semi-Markov model is chosen because this model can give a closer modeling approximation to the stochastic utilization of hospital resources than the Queuing theory and the Markov models.

The flow chart is found in figure 2 and Pseudo-code of capacity planning is given in figure 3.

Following are the basic assumptions for the Semi-Markov model in inpatient operations:

a) The transition probability only depends on the current treatment unit. How a patient enters this current treatment unit doesn't affect the transition probability. This assumption is the basic assumption of a Semi-Markov model. Without this assumption, inpatient operations analysis can't use Semi-Markov model.

b) After finishing medical treatment in unit i, a patient can't reenter this unit immediately. This assumption means that a patient can not transfer to another treatment unit until he finishes the required treatment.

c) When a patient enters one of the inpatient

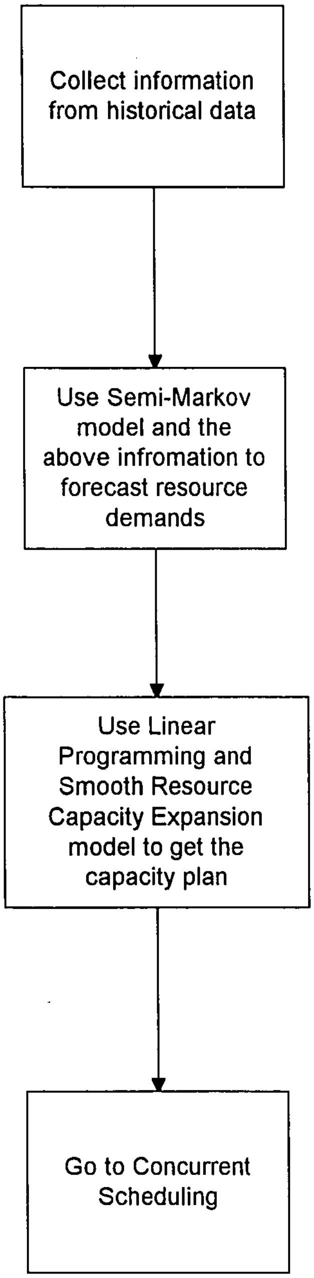

Figure 2: the procedure of Capacity Planning

The Capacity Planning Algorithm

Input: Historical data for process time, transition probability

Output: Expected demands on resources

Begin

Let $k = 0$

Ano-res Let $k = k + 1$

$h = 0$

Ano-DRG Let $h = h + 1$

P_h = the probability matrix for DRG_h

ET_{ijkh} = the first moment of estimated DRG_h process time in unit i before transfer to unit j for resource k

ET_{ijkh^2} = the second moment of estimated DRG_h process time in unit i before transfer to unit j for resource k

By the Semi-Markov model

Get ET_{tkh} = the estimated DRG_h total usage time for resource k per patient

Let P_h = the monthly number of DRG_h patients

$ET_{kh} = P_h \times ET_{tkh}$ = the monthly DRG_h time load for resource k

Let $ET_k = ET_k + ET_{kh}$

Figure 3 the capacity planning algorithm

If h = n and k < r, **Ano-res**

Else if h < n and k < r, **Ano-DRG**

end

n : the total no. of DRG

r : the total no. of resources

ET_k : the estimated annual required time for resource k

ETt

Figure 3 continued

operations units, he can immediately get required resources. The model doesn't consider waiting time in patient queue. This assumption has the Semi-Markov model consider only resource demands.

4.1.1 The Symbol Glossary For The Semi-Markov Model

p_{ij} = the transition probability that a patient will transfer from a treatment unit i to another treatment unit j or absorbing state j (including recovery, death, and outlier) after he receives a treatment in unit i. What follows is the structure of transition probability matrix:

$$P_{ij} = \begin{bmatrix} Q & R \\ 0 & I \end{bmatrix} \tag{1}$$

An example of transition probability matrix is found in table 1.

Q = the s x s transient state matrix

R = the s x (r-s) absorption state matrix

t_{ijk}^{1} = the first moment of the process time that a patient should stay in unit i, before he is transferred to another treatment unit or an absorption state j for resource k; unit (min)

t_{ijk}^{2} = the second moment of the process time that a patient should stay in treatment unit i, before he is transferred to another treatment unit or an absorption state j for resource k; unit (min2)

	emg.	reg.	surg.	icu	picu	outlier	recovery	death
emg.	0	0	0.98	0	0	0	0	0.02
reg.	0	0	1	0	0	0	0	0
surg.	0	0	0	0.08	0	0.01	0.9	0.01
icu	0	0	0	0	0.9	0.05	0	0.05
picu	0	0	0	0	0	0	1	0
outlier	0	0	0	0	0	1	0	0
recovery	0	0	0	0	0	0	1	0
death	0	0	0	0	0	0	0	1

the Q matrix

	emg.	reg.	surg.	icu	picu
emg.	0	0	0.98	0	0
reg.	0	0	1	0	0
surg.	0	0	0	0.08	0
icu	0	0	0	0	0.9
picu	0	0	0	0	0

the R matrix

	outlier	recovery	death
emg.	0	0	0.02
reg.	0	0	0
surg.	0.01	0.9	0.01
icu	0.05	0	0.05
picu	0	1	0

emg: Emergency admission
reg: Regular admission
surg: Surgery unit
icu: ICU
picu: Post ICU
outlier: outlier (absorbing state)
recovery: recovery (absorbing state)
death: death (absorbing state)

Table 1 An example of transition probability matrix

t_{ik} = the average process time in treatment unit i for resource k; unit(min)

$$t_{ik} = \sum_{j=1}^{s} t_{ijk}\ p_{ij}$$

t_{ik}^{n} = the nth moment of t_{ik} for resource k; unit(min^{n})

$$t_{ik}^{n} = \sum_{j=1}^{s} p_{ij}\ t_{ijk}^{n}$$

$M_k^{(1)}$ = the s × s diagonal matrix of (t_{1k}, t_{2k}, t_{3k}, . . . , t_{sk}) for resource k, such that

$$M_k^{(1)} = \begin{bmatrix} t_{1k} & 0 & 0 \\ 0 & t_{2k} & 0 \\ 0 & 0 & t_{3k} \end{bmatrix}$$

$M_k^{(2)}$ = the s × s diagonal matrix of (t_{1k}^2, t_{2k}^2, t_{3k}^2, . . . , t_{sk}^2) for resource k, such that

$$M_k^{(2)} = \begin{bmatrix} t_{1k}^2 & 0 & 0 \\ 0 & t_{2k}^2 & 0 \\ 0 & 0 & t_{3k}^2 \end{bmatrix}$$

M_k' = the s x s matrix of $p_{ij} \times t_{ijk}$ for resource k, such that

$$M_k' = \begin{bmatrix} P_{11} \times t_{11k} & P_{12} \times t_{12k} & P_{13} \times t_{13k} \\ P_{21} \times t_{21k} & P_{22} \times t_{22k} & P_{23} \times t_{23k} \\ P_{31} \times t_{31k} & P_{32} \times t_{32k} & P_{33} \times t_{33k} \end{bmatrix}$$

$R_k^{(2)}$ = the s x 1 column matrix of $(t_{1k}^2, t_{2k}^2, t_{3k}^2, \ldots, t_{sk}^2)$ for resource k, such that

$$R^{(2)} = \begin{bmatrix} t_{1k}^2 \\ t_{2k}^2 \\ t_{3k}^2 \end{bmatrix}$$

$N_k^{(1)}$ = the s x s matrix of the first moment of total process time in unit j for resource k before being discharged, under the premise that the first unit, where a patient enters into a hospital, is unit i; unit(min)

$$N_k^{(1)} = (I - Q)^{-1} M_k^{(1)} \qquad (1)$$

Let $\odot$ = a congruent matrix multiplication,

where

$$A \odot B = C$$

implies

$$a_{ij} \odot b_{ij} = c_{ij}$$

$N_k^{(2)}$ = the s x s matrix of the second moment of total process time in unit j for resource k before being discharged, under the premise that the first unit, where a patient enters into a hospital, is unit i; unit(min^2)

$$N_k^{(2)} = (I - Q)^{-1} [M_k^{(2)} + 2 I \odot (M_k' N_k^{(1)})] \qquad (2)$$

$V_k^{(1)}$ = the s × s matrix of the first moment of total process time for resource k before being discharged, under the premise that the first unit, where the patient enters into a hospital, is unit i; unit(min)

$$V_k^{(1)} = N_k^{(1)} \xi \qquad (3)$$

Let ξ = the s × 1 unit vector, such that for s = 3

$$\xi = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

$V_k^{(2)}$ = the s × s matrix of the second moment of total process time for resource k before being discharged, under the premise that the first unit, where the patient enters into a hospital, is unit i, unit(min^2)

$$V_k^{(2)} = (Q - I)^{-1} [R^{(2)} + M' V^{(1)}] \qquad (4)$$

F_k = the 1 × s probability vector which represents the probability that the first unit for resource k, where a patient enters into a hospital, is unit i, such that for s = 3, we might have from historical data

$$F_k = \begin{bmatrix} 0.8 \\ 0.2 \\ 0 \end{bmatrix}$$

ET_{ik} = the scaler which represents the expected total process time in a hospital per admission of DRGi type for resource k.

$ET_{ik} = F_k \; V_k^{(1)}$ (6)

λ_i = the monthly arrival frequency of DRGi patients

ET_{tk} = the cumulative expected monthly process time for resource k

ET_{tik} = the expected monthly process time of DRGi patients for resource k

$ET_{tik} = \lambda_i \times ET_{ik}$ (7)

n = the total number of DRGs

S_{sd} = the standard deviation of data group S

$S_{sd} = [S^{(2)} - S^{(1)} \odot S^{(1)}]^{0.5}$ (5)

To get the cumulative expected monthly process time for resource k, what follows is the formula for the cumulative expected monthly process time for resource k

$$ET_{tk} = \sum_{i=1}^{n} ET_{tik}$$

4.1.2 The Procedure To Get The Expected Monthly Demands On Each Resource.

Step 1 :

Get the transition probability from historical data

Step 2 :

Get the first moment of the process time that a patient should stay in treatment unit i, before he is transferred to another treatment unit j or a discharged state j.

Get the second moment of the process time that a patient should stay in treatment unit i, before he is transferred to another treatment unit j or a discharged state j.

Step 3 :

By formula (1), we can get $N^{(1)}$.

Step 4 :

By formula (2), we can get $N^{(2)}$. Then by formula (5), we can get the standard deviation of total process time in unit j before being discharged.

Step 5 :

By formula (3), we can get $V^{(1)}$.

Step 6 :

By formula (4), we can get $V^{(2)}$. Then by formula (5), we can get the standard deviation of total process time before being discharged.

Step 7 :

By formula (6), we can get the expected total process time per admission.

Step 8 :

By formula (7), we can get an expected monthly demand of a particular resource for a particular type of DRG.

An example procedure of Semi-Markov model is given in figure 4. The example is the monthly demand for surgeons by DRG1 patients.

A more detailed procedure of the above example is in appendix 1.

The process time for each treatment unit per DRG1 patient is in appendix 2; the process time for each treatment unit per DRG2 patient is in appendix 3; and the process time for each treatment unit per DRG3 patient is in appendix 4. The transition probability for each DRG is in appendix 5. The above data can evolve resource demands for

(unit.min)

Step 1 : get the transition matrix

	emg.	reg.	surg.	icu	picu	outlier	recovery	death
emg.	0	0	0.98	0	0	0	0	0.02
reg.	0	0	1	0	0	0	0	0
surg.	0	0	0	0.08	0	0.01	0.9	0.01
icu	0	0	0	0	0.9	0.05	0	0.05
picu	0	0	0	0	0	0	1	0

Step 2 : get the first moment of process time

	emg.	reg.	surg.	icu	picu	outlier	recovery	death
emg.	0	0	18	0	0	0	0	18
reg.	0	0	0	0	0	0	0	0
surg.	0	0	0	440	0	440	440	440
icu	0	0	0	0	0	0	0	0
picu	0	0	0	0	0	0	40	0

get the second moment of process time

	emg.	reg.	surg.	icu	picu	outlier	recovery	death
emg.	0	0	421	0	0	0	0	421
reg.	0	0	0	0	0	0	0	0
surg.	0	0	0	202500	0	202500	202500	202500
icu	0	0	0	0	0	0	0	0
picu	0	0	0	0	0	0	2070	0

Step 3 : use formula (1) and get N(1)

	emg.	reg.	surg.	icu	picu
emg.	18	0	431.2	0	2.8224
reg.	0	0	440	0	2.88
surg.	0	0	440	0	2.88
icu	0	0	0	0	36
picu	0	0	0	0	40

Step 4 : use formula (2) and get N(2)

	emg.	reg.	surg.	icu	picu
emg.	421	0	198450	0	146.0592
reg.	0	0	202500	0	149.04
surg.	0	0	202500	0	149.04
icu	0	0	0	0	1863
picu	0	0	0	0	2070

Figure 4 **An example of Semi-Markov model**

use formula (5) and get Nsd

	emg.	reg.	surg.	icu	picu
emg.	9.848858	0	111.8774	0	11.75131
reg.	0	0	94.33981	0	11.86363
surg.	0	0	94.33981	0	11.86363
icu	0	0	0	0	23.81176
picu	0	0	0	0	21.67948

Step 5 : use formula (3) and get V(1)

emg.	452.0224
reg.	442.88
surg.	442.88
icu	36
picu	40

Step 6 : use formula (4) and get V(2)

emg.	217125.6
reg.	205183.4
surg.	205183.4
icu	1863
picu	2070

Step 7 : use formula(6) and get the expected total process time

ETti 96.889

Step 8 : use formula (7) and get the monthly demand

ETi 1937.777

emg: Emergency admission
reg: Regular admission
surg: Surgery unit
icu: ICU
picu: Post ICU
outlier: outlier (absorbing state)
recovery: recovery (absorbing state)
death: death (absorbing state)

Figure 4 **continued**

each kind of resource.

Because patient bed's time unit is "day" and other resource's time unit is "minute", results from the Semi-Markov model are separated into two parts: one is "patient beds" and the other is "other resources". Demands for patient beds are given in appendix 6. Demands for other resources can be found in appendix 7 (per "regular admitted" particular type of DRG patient), appendix 8 (per "emergency admitted" particular type of DRG patient), and appendix 9 (per "admitted" particular type of DRG patient)

4.2 The Capacity Planning Mathematical Model

To prepare for future capacity expansion, demand growth for inpatient operations must be analyzed in advance. Freidenfelds (1981) provided models for capacity expansion for many types of demand growth. When the demand growth type is decided, forecasted demands (or patient arrival frequency) for the following periods (here, a planning period is a month) in the current planning horizon (here, is a planning horizon is a year) will be determined by using historical data.

The model employed will use the monthly demands of various DRG types on the hospital system which are translated into loads on individual resources as determined by the Semi-Markov model, discussed earlier, to determine an optimal capacity plan. This is a Multi-Production Linear Programming (MPLP) Model with dynamic "job" (patient) arrival and linear resource costs as described in Johnson and Montgomery (1974). The major concern in capacity planning is to minimize resource costs and meet resource demands.

4.2.1 The Symbol Glossary For The Capacity Planning Mathematical Model

N_m : the number of weekdays in the current planning period (here a planning period is a month)

N_{ri} : the number of units of regular resource i planned to be employed in the current planning period

L_{ri} : the regular shift length of resource i; unit(hr)

L_{oi} : the monthly overtime planned for resource i; unit(hr)

L_{ci} : the monthly allocation of workload to outside contract resource i; unit(hr)

C_{ri} : the hourly costs of a regular shift for resource i; unit($/hr)

C_{oi} : the hourly costs of overtime for resource i; unit($/hr)

C_{ci} : the hourly costs of outside contract resource i; unit($/hr)

θ : the maximum legal value of "L_{oi} divided by $(N_{ri} \times L_{ri})$", this value can be found in the Union Contract.

D_i : the monthly demand for resource i; unit(min/month)

n : the total number of resource categories

4.2.2 The Capacity Planning Mathematical Model

Resources discussed in this capacity planning mathematical model are medical staffs and health care facilities not including the pharmacy, cafeteria, and other non-medical resources. There are some characteristics of the industrial multiproduction model that are specialized for this hospital application. For example, equipment doesn't have overtime and is available around the clock. Thus, the overtime costs of equipment are the same as the regular time costs, but human labor is a different story. The overtime costs of personnel are higher than the regular costs. Besides, the equipment's overtime is the same as its operators (or medical staffs), because equipment can not function without operators.

The objective of capacity planning is to save (minimize) costs in utilizing resources and meet demands. So the objective function is:

$$Min \sum_{i=1}^{n} \left[\left(L_{ri}\; N_m\; C_{ri} \right) \times N_{ri} + C_{oi} \times L_{oi} + C_{ci} \times L_{ci} \right]$$

Subject to:

$$(L_{ri}\ N_m) \times N_{ri} + L_{oi} + L_{ci} \geq D_i$$

$$i = 1, 2, 3, \ldots, n \qquad (8)$$

In constraint (8), the total capacity for a particular resource must not be smaller than the demand for this resource in the current month.

$$(\theta\ L_{ri}\ N_m) \times N_{ri} \geq L_{oi}$$

$$i = 1, 2, 3, \ldots, n \qquad (9)$$

In constraint (9), the maximum amount of overtime for a particular resource can not be larger than the value that what is legally allowed in the union contract for unionized employees. This is set up for labor safety and rights.

Model parameters:

$$N_m,\ L_{ri}\ ,\ C_{ri}\ ,\ C_{oi}\ ,\ C_{ci}\ ,\ \theta\ ,\ D_i \geq 0$$

Model decision variables:

$$N_{ri},\ L_{oi},\ L_{ci} \geq 0$$

N_{ri}, L_{ri}, N_m, C_{ri}, L_{oi}, C_{oi}, L_{ci}, C_{ci}, and θ are non-negative real numbers. N_{ri} is an integer because N_{ri} is the countable capacity of regular resources. The other numbers, such as θ, are positive real numbers.

In this capacity planning model, C_{ri}, C_{oi}, C_{ci} are cost factors which are known; N_m is the number of weekdays in current month; θ is a maximum legal value which also is a fixed value; and demand for resource i is obtained from the Semi-Markov model.

To get the cumulative expected monthly process time for resource i, following is the formula for the cumulative expected monthly process time for resource i

$$D_i = \sum_{j=1}^{n} ET_{tij}$$

ET_{tij} : the expected monthly process time of DRG_j patients for resource i; unit(min/month)

D_i : the monthly demand for resource i; unit(min/month)

So the variables in objective function, N_{ri}, L_{oi}, and L_{ci}, can be assessed easily by using the Lindo linear programming package.

What follows are some conditions for inpatient operations:

1) $(N_m L_{ri}) \times C_{ir} < (N_m L_{ri}) \times C_{oi} < (N_m L_{ri}) \times C_{ci}$: this condition means capacity planning will first allocate demands to regular resources as much as possible.

2) $C_{oi} < C_{ci}$: this condition means that capacity planning then will allocate demand to overtime as much as possible. As stated in constraint (9), there are maximum limits for overtime, so overtime can take a certain amount of demand. The rest of the demand will be allocated to an outside contract resource or another regular resource having lower costs.

Capacity plans for all 12 months in the next planning horizon (a year) are determined using a Parametric Production Planning model [Hax and Candea (1984)] which employs a smoothing work force decision rule. This model will smooth monthly inpatient operations resource adjustments and avoid fierce resource capacity fluctuation. Fierce resource capacity fluctuation will deteriorate service quality and increase personnel training costs, hiring costs, and layoff costs. The rule is the following:

$$N_{t,k} = N_{t-1,k} + A\ (\ N_{d,k} - N_{t-1,k}\ )$$

$N_{t-1,k}$: the existing capacity for a particular resource k

$N_{t,k}$: the planned capacity of the current planning period (or month) for a particular resource k

$N_{d,k}$: the desired number of units of a particular regular resource k for the current planning period

A : coefficient determining the fraction of desired change of resource to be implemented in the current planning time horizon; $0 \le A \le 1$. When A is close to 1, $N_{t,k}$ will be closer to $N_{d,k}$ and the performance of Parametric Production Planning model will be more responsive to demand changes; when A is close to 0, $N_{t,k}$ will be close to $N_{t-1,k}$ and may not meet the demand of the coming expansion period. But to avoid capacity shortage, a larger A is suggested in earlier runs, then gradually decrease A to an optimal value which will save costs and satisfy demands for inpatient operations.

$$N_{d,k} = \sum_{x=1}^{12} b_x\, N_{I,\,[t-1]+x,k}$$

b_x : weighting coefficient for the next xth demand forecast period, its formula is the following:

$$b_x = \frac{B^x}{\sum_{x=1}^{12} B^x}$$

B^x : B is a coefficient between 0 to 1; B^x is B to the xth power. Using this coefficient, the farther out in the planning period, the lesser the effect of N_t. In addition, when B is close to 1, every planning period has similar affect on $N_{d,k}$ and a very smooth capacity plan will be determined, but $N_{d,k}$ sometimes will not be able to meet a suddenly surging demands; when B is close to 0, the fluctuation of capacity plan will be more severe with associated increasing costs. To meet the coming demands, a small B is suggested in earlier runs, then gradually increase B to an "optimal" value as agreed to hospital administration after reviewing the resulting plans.

$N_{r,[t-1]+x,k}$: optimized regular resource allocation which is determined by the linear programming capacity planning model and is exactly the same as N_{ri} for resource i in the xth planning period in the future. Using the above approach, a stable capacity plan can be determined.

This capacity plan would be analyzed by sensitivity analysis to assure the plan saves costs and satisfies other management objectives. Winston (1994) gives a simple and clear reference. Besides sensitivity analysis, the severity of diseases should also be discussed by experienced senior administrative staff to avoid deteriorating quality of inpatient operations. If the administration feels the results of capacity planning will deteriorate the quality of inpatient operations, because of capacity shortage, increasing A or decreasing B would be suggested. If it is thought that the result of capacity planning will increase capacity idle time, decreasing A or increasing B will be suggested.

4.3 The Concurrent Scheduling Model:

In inpatient operations, patient "paths" through the hospital system are very versatile. Each patient has his own path. So before beginning a discussion of Concurrent Scheduling, the environment of the inpatient operations is investigated. In manufacturing, there are many types of environments, such as Job Shop, Batch Shop, and Flow Shop. Among these environments, the inpatient operations is similar to a Job Shop, because the basic characteristic of Job Shop problems is that every job has its particular flow of operations. In fact inpatient operations is more complicated than regular Job Shops, because the total process time of patients is shorter and the arrival frequency of patients is higher than those of a typical Job Shops.

Now, the way to choose the most appropriate scheduling method will be discussed. Every method has its particular strong points and weak points for different type of work shop. "Linear Programming" and "Branch and Bound" work well for the Flow Shop and the Batch Shop style problems, if the complexity of these problems is not too great. But Job Shop problems, even small problems, are NP-complete problems and are usually too large and too complex for these two methods. Fortunately, there are many heuristic methods which can solve "Job Shop" problems inexpensively and efficiently.

These heuristic methods will be discussed separately.

Morton and Pentico (1993) noted that an iterated myopic policy uses a recursive formula to converge to an objective function value. After a number of iterations, if the objective function value has no improvement, then the objective value and scheduling policy are determined. The iterated myopic method can function very well when there are no unforeseen arrivals of patients or hospitals can correctly predict these emergency patients during each time interval. But in the real world, there are always some emergency admission cases happening during each time interval. These cases will disrupt existing optimal schedules and lead to inferior outcomes.

To avoid idle resources, classic dispatch methods will allocate resources to the current highest priority jobs. Unfortunately, these methods do not guarantee an optimal result. But some "mixed" Dispatch methods can get better result than the Classic Dispatch methods. Conway (1965) noted that some mixed simple dispatch methods can outperform the robust Shortest Process Time (SPT) method.

Stecke and Solberg (1981) noted that SPT/TOT method outperformed (or tied with) most of the classic Dispatch Heuristic methods at an assembly factory. "TOT" means the total process time for the job. A job that has a smallest value of operating time divided by the total processing time for the job will have the highest priority.

In Stecke's model, the priority rule is

$$\pi_{ijk} = T_{ijk} / T_i$$

A smaller π_{ijk} has a higher priority. Because in the Concurrent Scheduling presented in this thesis, a larger priority value has a higher priority, Stecke's model is made to conform by inverting the formula:

$$\pi_{ijk} = T_i / T_{ijk}$$

π_{ijk} = the priority value for job i in a job queue k waiting for resource j. Here a larger π_{ijk} has higher priority.

T_i = the total process time of job i

T_{ijk} = the process time of job i in a job queue k waiting for resource j

When resource j is available, any job, whose priority value is highest across all job queues waiting for resource j, will get resource j.

The SPT/TOT method takes advantage of SPT, which reduces total waiting time, and avoids long jobs waiting too long by utilizing TOT.

In inpatient operations, the patient's process time in a single treatment unit can't be exactly predicted. So patient's total process time is much more difficult to predict exactly. So SPT/TOT method is not appropriate for inpatient operations.

To determine a more appropriate mixed heuristic method, the correlations between time (including treatment time and waiting time) and costs (including waiting costs and treatment costs) must be carefully investigated. For example, in inpatient operations, there is a "penalty cost" that increases exponentially according to the length of waiting time.

According to the figure 5, disease severity deteriorates exponentially with time for acute diseases and terminal diseases. The longer these patients stay, the worse their physical conditions tend to be. The more severe their physical conditions, the more difficult the therapy. So "penalty" costs associated with delay exist. The resource for the figure 5 is Hornbrook (1983).

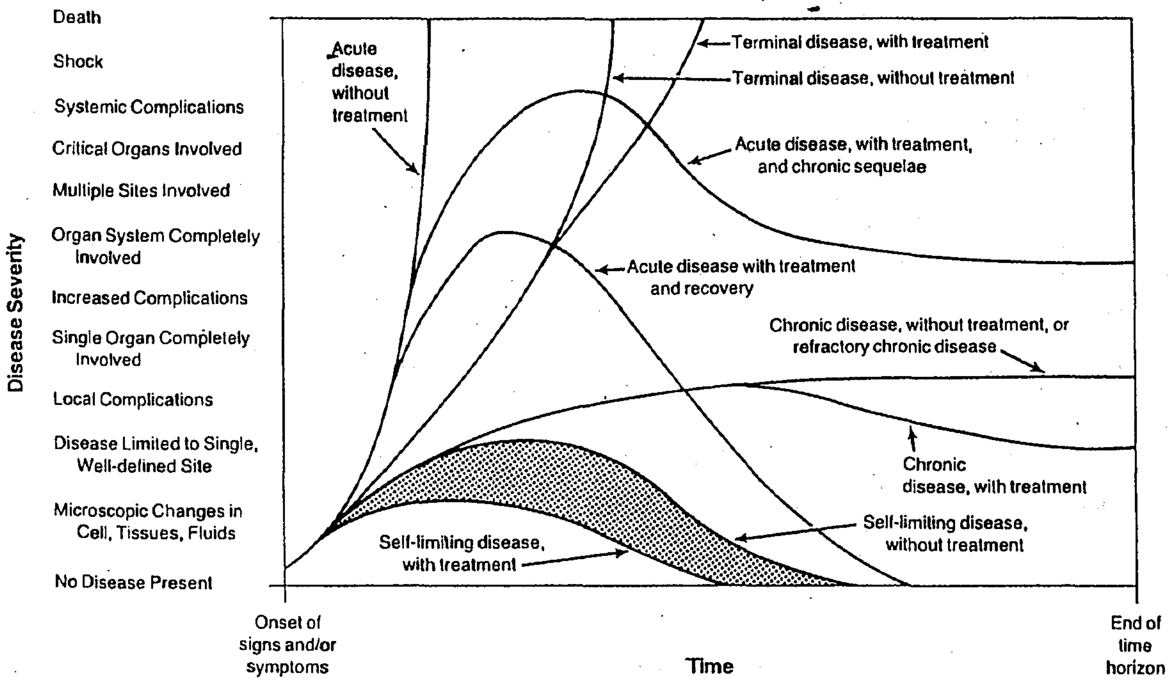

Figure 5 **types of disease severity profile, with and without treatment intervention**

4.3.1 The Definition Of The Concurrent Scheduling Model:

The first step of this analysis is to study the cost factors. There are 7 categories of inpatient operations costs:

1) direct nursing care costs,

2) indirect nursing care costs,

3) room and bed costs,

4) variable overhead costs,

5) fixed overhead costs,

6) ancillary costs, and

7) direct physician costs.

1) The direct nursing care costs include:

a) average hourly nursing salary,

b) benefits which are present as the percentage of average salary, and

c) coverage cost which is a function of the average employee absenteeism per year and is determined as a percentage of total working days.

2) The indirect nursing care costs include:

a) nursing unit support services cost(including head nurse, charge nurse, and ward secretary costs) and

b) nursing unit administrative support service cost(including costs of directors of nursing, associate directors, in service educators, infection control coordinators, nursing staff secretarial staffs, staffing office personnel, and shift supervisors).

3) Room and bed costs include:

a) depreciation and interest on buildings and other fixed assets,

b) capital related to major movable assets,

c) maintenance and repairs,

d) plant operations,

e) laundry and linen, and

f) house keeping.

4) Variable overhead costs include:

a) dietary and

b) medical records and library.

5) Fixed overhead costs include:

a) administration and general overhead,

b) cafeteria,

c) central services,

d) pharmacy, and

e) social service.

6) Ancillary costs include:

a) X-ray,

b) radiology,

c) lab,

d) supplies,

e) drugs,

f) ambulance,

g) surgery, and

h) others.

7) Total Physician direct care costs include:

a) average hourly physician salary,

b) benefits which are present as the percentage of average salary, e.g., 16 %, and

c) coverage costs.

The sum of item (1), (2), (3), and (4) is the regular patient room medical costs. Whenever patients are admitted into hospitals, the costs are proportional to LOS in patient rooms.

Item (5), a fixed cost is spread over the number of admitted patients over a one month period (for example).

Item (6) is charged according to the kinds of ancillary services patients receive.

Item (7) is charged according to the service duration of patients.

Detailed materials about how to get these costs can be found in Vincent (1991).

The second step of this analysis is resource analysis and resource grouping.

1) Identify all resources for hospital inpatient operation.

a) physicians - internists, surgeons, obstetricians, gynecologists, psychiatrists, anesthetists, pathologists, therapists, etc...

b) health facility specialists - radiologists, anesthesiologists, and X-ray technicians.

c) health equipment and related labs - labor rooms, delivery rooms, chemical and bacteria labs, X-ray labs, therapy rooms, electrocardio labs, nurseries, recovery rooms, operating rooms, and emergency rooms.

d) Room and Beds - internal medical patient beds, surgery patient beds, intensive care unit beds, and emergency room beds.

2) Group resources which are ready at the same time and at the same location into a resource group. For instance, an anesthetist, several special nurses, and equipment can be grouped into an operating room. This step will make the Concurrent Scheduling method easier to implement.

3) When two or more resources are required for a certain service, set the resource, whose total process time is longest, as the major resource. Then set the resource, whose total process time is second to longest, as the secondary resource. Basically in inpatient operations, there is no obvious third resource. Most of resources can be grouped into a major resource or a secondary resource. In inpatient operations, there are two major medical staffs: physicians and nurses. Physicians are independent, not belonging to a specific working place, but nurses have to work at a specific treatment unit and provide fixed skill level services. Nurses are not usually abruptly reassigned to work at another treatment unit, provide higher skill level or different skill services. For example, special nurses in operating rooms can not work at ICU and ICU nurses can't work in operating rooms. If there is a third resource, this resource is the one whose total process time is the third longest. The secondary resource (or third resource) has to cooperate with the major one. This means when the major resource is available, the highest priority patient will get the major resource. Then this patient automatically has the highest priority to get the secondary resource (or third resource). In inpatient operations, the most obvious example is surgery. In the surgery operation, the operating room is the major resource and the surgeon is the secondary resource. According to

Statistical Reference Index (1993) referring to hospitalization, the weekly average number of surgery hours for a surgeon is around 8 hours. An operating room works at least 8 hours per day. In fact, hospitals would like to increase operation room utilization as much as possible.

4) Patients can be divided into two groups according to how patients choose their physicians.

a) A patient, who accepts only one particular physician to take care of him: when this patient has already chosen his physician, the daily patient time load of the physician increases by this patient's daily process time, such as the process time of hospital rounds. In addition, if the patient needs surgery, the surgery time will be added to this physician's total daily inpatient time.

b) A patient, who will accept any physician to take care of him: this kind of patient will help to balance the work load of physicians. When this kind of patient enters a hospital, he will be assigned to a physician whose daily patient time load is smallest. When all physician's work loads are close to each other, a particular physician's patients are less likely to wait too long for treatment.

4.3.2 The Concurrent Scheduling Model Procedure

1) When a patient is ready for a specific treatment in his DRG, if the required resources are available, the patient receives treatment; go to step(3). Else, go to step(2)

2) The patient enters the service queue for required resources.

3) Provide the treatment: when the treatment is finished, the patient releases all resources. The patient will stay in the same unit for a period of time for treatment and observation, then is transferred to another medical unit or discharged from the hospital according to the transition probability matrix for the patient's DRG. At last, he goes to step (1), if he is transferred to another medical unit; else, he leaves this treatment unit and is discharged from the hospital. When he releases all resources, a check is made as to whether the treatment given at the present medical unit needs one kind of resource or two kinds of resources. If the treatment needs only one kind of resource, go to step 3.1). Else, go to step 3.2).

3.1) Check whether there are any patients waiting in the patient queue. One of the following conditions could hold.

a) There is no patient in this queue. The resource waits for a new patient to arrive.

b) There is at least one patient in this queue. Determine the highest priority patient by Concurrent Scheduling priority rule and allocate the resource to the patient; go to step 3).

3.2) Check whether there are any patients waiting in the patient queue. One of the following conditions could happen.

a) There are no patients in this multi-resource queue. The major resource waits for a new patient to arrive. Allocate the secondary resource to a patient who has first priority to get it.

b) There is only one patient in this multi-resource queue. Allocate the major resource to the patient and he has the first priority to get the rest of required resources. When he gets all required resources, go to step(3). If the secondary resource of this patient isn't that of the last patient, the secondary resource of last patient is allocated to another patient who has first priority.

c) There are two or more patients in the service queue. There are several steps for this situation.

c.1) Find these patients' physicians.

c.2) Find the physician whose daily patient time load is largest among the above physicians.

c.3) Find the highest priority patient among this physician's patient(s). The major resource is allocated to this patient and he automatically has the first priority to get the rest of required resources. When he gets all required resources, go to step(3). If the secondary resource of this patient (here the secondary resource is physicians) isn't that of last patient, the secondary resource of last patient is allocated to a patient who has first priority to get the secondary resource.

Figure 6 summarizes patient path logic as presented above. The pseudo-code of Concurrent Scheduling is given in figure 7.

The priority rule used for the Concurrent Scheduling model is defined as:

$$\pi_{ilk} = [\ (\ P_{il} - C_{kil}\ ) + C^{p}_{il} \exp(T^{n} - T^{a}_{i})/T^{p}_{i}]\ /\ T_{il}$$

π_{ilk} = the priority value of a DRGi patient who waits in a queue for the major resource k utilized in treatment l within DGRi

P_{il} = the treatment l price for a DRGi patient

C_{il} = the treatment l costs for a DRGi patient

C^{p}_{il} = the penalty costs of treatment l for a DRGi patient; its unit is "$".

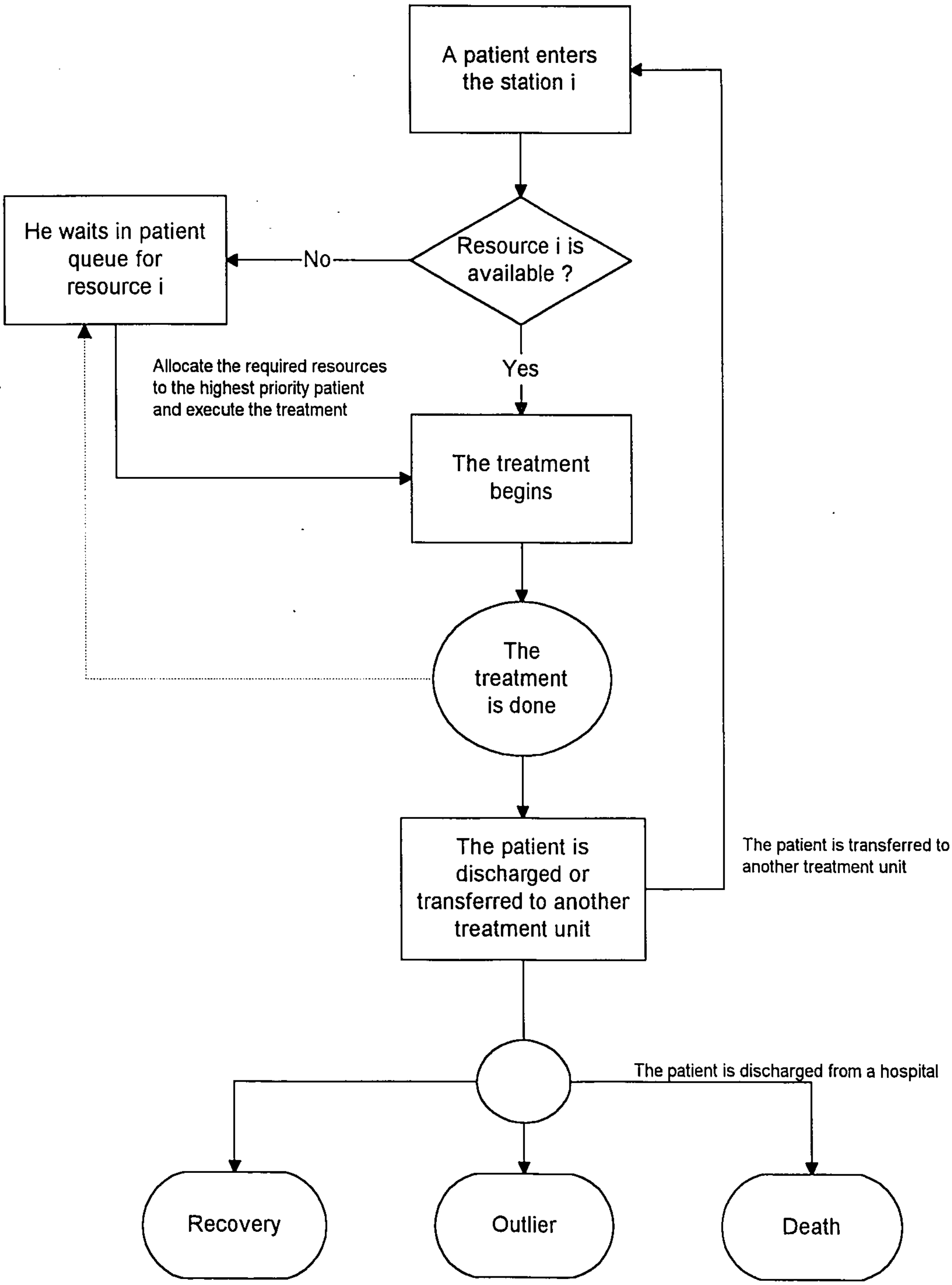

Figure 6: the procedure of Concurrent Scheduling

The concurrent Scheduling Algorithm

Input : Inpatients

Output : Recovery , death , or outlier

Begin

Next-stat Enter station i

If $R(i)_k=1$ go to **Execute**

Else go to **Wait**

Execute Let $R(i)_k= 0$

Delay [process-time(i)]

Decide next station by historical transition probability

If this treatment unit requires two or more resources, send a signal to **Critical**

Else, send a signal to **find-best**

Go to **Next-stat**

Critical Find all physicians who have patients in $queue_i$

Find the physician whose daily work load is highest among these physicians

Allocate resources to the patient who has the highest priority among this physician's patients

If there is no patient in the $queue_i$,

let $R(i)_k=1$

End

Figure 7 the Concurrent Scheduling algorithm

Find_best Find the highest patient in $queue_i$

Allocate resource to this patient and send this patient to **Execute**

If there is no patient in the **$queue_i$**,

let $R(i)_k=1$

End

Wait wait in the $queue_i$ for the required resource(s)

End

$R(i)_k$: if the resource number k for station i is available , then $R(i)_k = 1$: else, $R(i)_k = 0$

$queue_i$: the patient queue in treatment unit i

process-time(i) : the process time for patients in treatment unit i

priority value can be determined by Concurrent Scheduling priority rule

Figure 7 continued

T_{il} = the expected process time of the treatment l for a DRG_i patient

T^n = the current time

T^a_i = the arrival time of a DRGi patient at the current treatment unit (Information Technology can help to manage this.

T^p_i = the time penalty factor for a DRG_i patient

This rule is a mixed dispatch heuristic rule. This rule merges the cost factor, price factor, time factor, and disease severity factor together.

The penalty cost factor will make inpatients have higher priority than outpatients. In addition, more severely ill patients will have a larger penalty cost factor.

The price factor will make more profitable patients have higher priority than less profitable patients and help hospitals enhance positive cash flow.

The disease severity factor, which is represented by the time penalty factor and the penalty cost, will keep severely ill patients from waiting too long. In the Concurrent Scheduling model, the more severe the disease is, the shorter the disease time interval is, the higher the penalty cost is. So, severely ill patients will have relatively higher priorities as time passes.

The process time factor will help hospitals choose the most profitable / unit time patient.

To examine the above priority formula, relations across all patient queues will be discussed. In inpatient operations, there are three kinds of competitions across patient queues.

1) the competition between inpatients and outpatients for the same resources, such as for diagnostic test labs.

2) the competition among inpatients who need surgery operations.

3) the competition among inpatients who need non-surgery treatment and need the same resources, such as an ICU and a therapy room.

The priority rule can effectively choose the "best" patient to associate with resources in these kinds of competitions.

For the first kind of competition, inpatients will have higher priority, because the regular treatment costs and the penalty costs of inpatients are much higher that those of outpatients.

For the second kind of competition, the rule will try to find the best choice by trading off profits and the illness severity.

For the third kind of competition, like the second kind of competition, the most important patient will have the first priority.

5. An Experimental Investigation Of The Concurrent Scheduling Heuristic:

There are two kinds of admissions in inpatient operations - the emergency admission and the regular admission.

5.1 The Emergency Admitted Inpatient Therapy Procedure:

1) When a patient enters a hospital, he will be admitted into the hospital immediately.

2) After being admitted, he will be assigned any available physician in the emergency room. If he needs a specialist, he can pull in this physician with the highest priority from another treatment unit. An emergency patient will have the highest priority to get resources, except if other more severely injured emergency patients are admitted.

3) When the patient is stable, his physician is released.

4) The patient will stay in the emergency room for a period of time to have the appropriate treatment and observation. When this is done, this patient has the first priority to get a regular bed and is transferred to a

regular treatment unit.

5) Every hospital will have its own patient allocating policy to assign this patient. Basically the rotating policy, "physicians will get new patients by rotating", is the most common policy in hospitals.

6) The patient requires further treatment. Under the hospital's service policy (such as FCFS method), the patient gets the required resources and the treatment begins; go to step(7). Else, he enters the patient queue for the required resources.

7) After this treatment, the doctor will decide whether the patient requires another medical treatment or is discharged from the hospital (not including death). If the patient is dead, he will be automatically discharged.

8) If the patient is sent to another unit, he will go to step(6).

9) If the patient is discharged, his physical condition can be one of following:

a) recovery (a good scenario),

b) outlier (the patient's condition becomes worse and is beyond the hospital's control, he is sent to another higher level hospital, a bad scenario), and

c) death (the worst scenario).

The graphic summing of regular admission is given in figure 8 and that of emergency admission is given in figure 9.

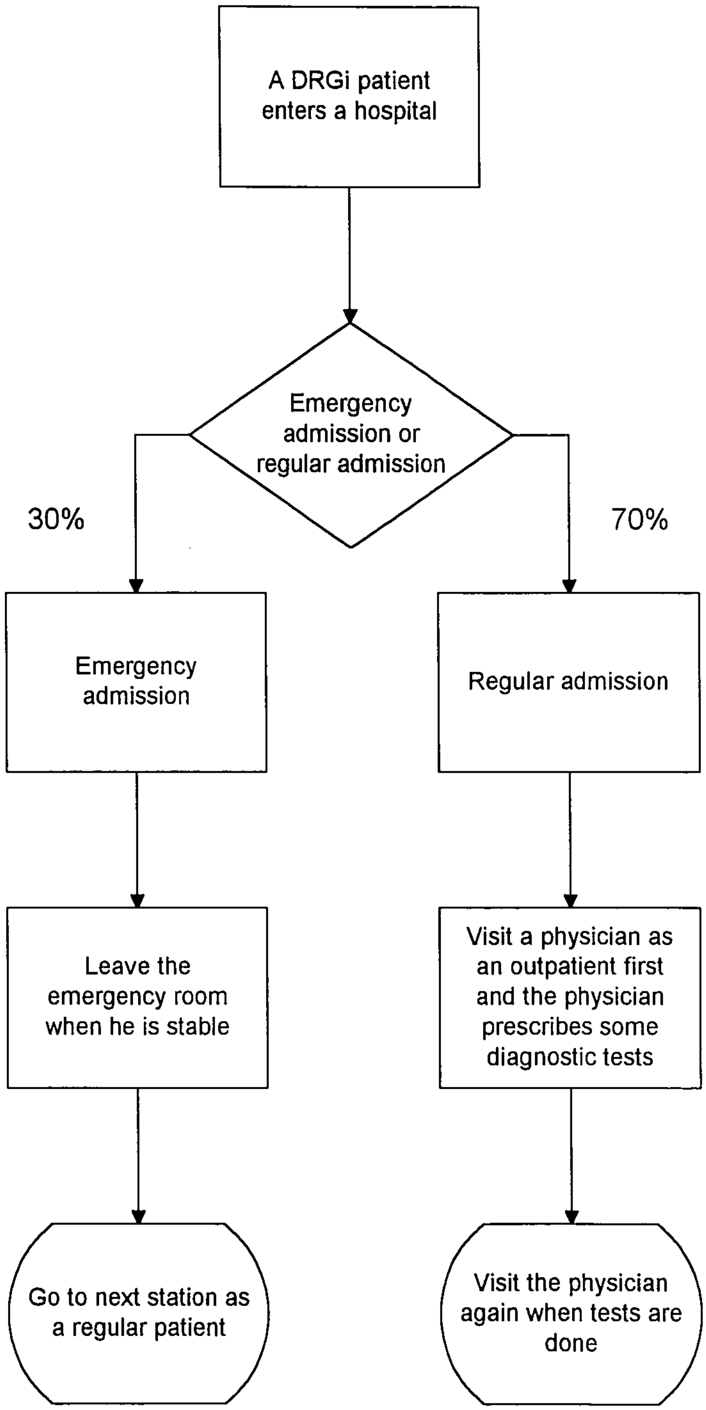

Figure 8: the procedure of admission

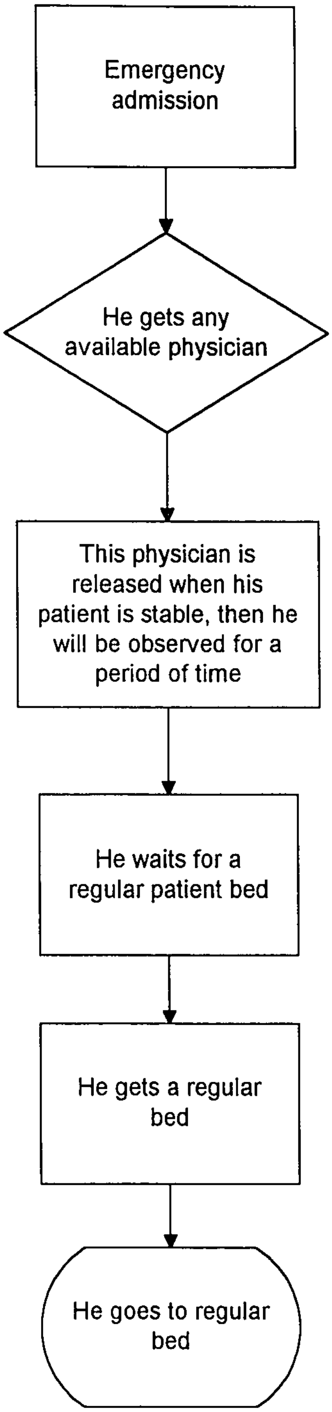

Figure 9: the precedure of emergency admission

5.2 The Regularly Admitted Inpatient Therapy Procedure :

1) Patients can be defined into two group.

Group 1: the patients who have already chosen a specific physician before entering this hospital.

Group 2: the patients who will take any physician as their physician. Every hospital will have its own patient allocating policy. The "rotating" policy is the most common policy in hospitals.

2) First, the patient will wait for his specific physician to perform pre-diagnosis. Then, his physician will prescribe diagnostic tests for him. If the specific physician is not available, he will enter the pre-diagnosis queue associated with his physician.

3) After pre-diagnosis, procedures are conducted at the appropriate laboratories. These procedures only take a short time (for example 10 minutes to take a X-ray picture), but the results of inspections will take a longer time (for example 4 hours to develop X-ray films).

4) According to the result of tests, the doctor will indicate this patient's DRG and assign him to an appropriate treatment unit.

5) Under the hospital's service policy (such as FCFS method), he either gets required resources or is on the patient queue for required resources. If he gets the required resources, the treatment begins; go to step (6). Else, he enters the patient queue for the required resources.

6) After receiving treatment, the doctor will decide whether he requires further medical treatment or is discharged from the hospital (not including death). If he is dead, he will be automatically discharged.

7) If the patient is sent to another unit, go to step(5);

8) If he is discharged, his physical condition can be

a) Recovery (a good scenario)

b) Outlier (the patient's disease becomes worse and is beyond the hospital's control and is sent to another higher level hospital, a bad scenario), and

c) Death (the worst scenario).

Regular admission procedure is depicted in figure 10.

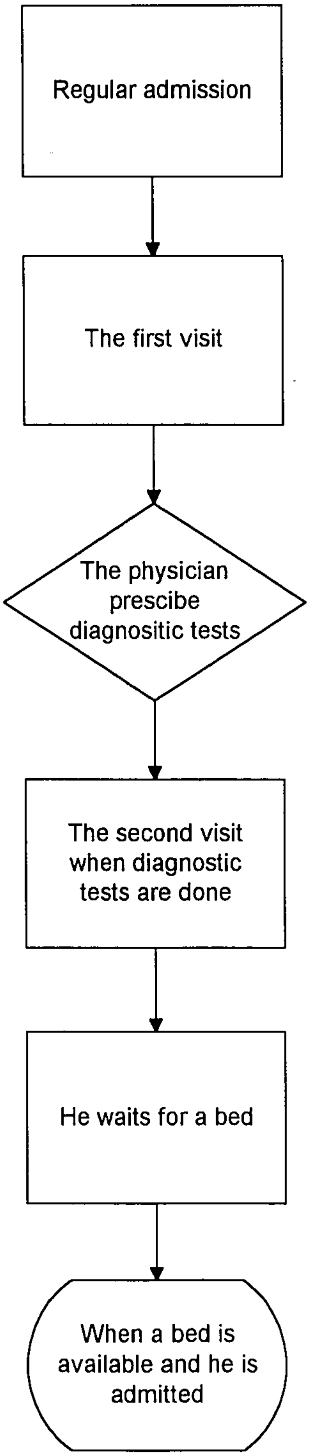

Figure 10: the procedure of regular admission

5.3 Experiment Definition And Assumptions:

In the real world, a metropolitan hospital may have nearly a thousand of beds and hundreds of medical staffs. A large model just has more resources and more treatment units than a simple one. So if a simplified model with the resource and patient interaction complexities in place can prove that the heuristic method works well, then there is a good reason to believe that it will work well in the larger setting. That is why a simplified example is explored here.

In the following are the basic definitions and assumptions for this simple inpatient operations.

1) There are only 3 kinds of DRGs: DRG1, DRG2, and DRG3.

2) Each physician, surgeon, and anesthetist works 8 hours per day (not including overtime and one hour lunch off) and 5 days per weeks. On weekends, at least one of the physicians per treatment unit has to stay in the hospital for some emergency.

3) Each nurse works one shift per day. There are three nursing shifts per day.

4) 70% of admitted patients have decided upon a specific physician before they enter a hospital. The rest

are assigned to physicians according to hospital patient allocating policy.

5) The diagnostic testing function of the hospital includes four units : a X-ray lab, a Magnetic Resonate Image (M.R.I.) lab, a chemical and bacteria lab, and an electrocardio lab. These lab operators work 8 hours per day (not including overtime and one hour lunch off), 5 days per week, for one shift.

6) Support services(including dietary services, laundry, pharmacy, janitor,...) are not considered here.

7) The process time in each treatment unit is normally distributed. The process time of outpatient visits and hospital rounds for each inpatient is fixed.

8) The interarrival distribution of DRGi patients is a negative exponential and is independent of other DRGs.

9) Each treatment has its own costs and price.

10) When a patient needs multi-resources to execute a treatment, this treatment can't begin if the patient doesn't get all required resources.

12) The outputs of inpatient operations is measured by throughput of all DRGs.

13) The most important assumption : the process time can be correctly predicted with regard to its mean and variance.

The simple inpatient operations will have 1 operating room, 3 surgeons, and 36 beds.

A warm-up period is the period of time in which a system is just beginning to operate and patient flow doesn't reach its stable, steady-state operation. To assure the results of the simulation are from a stable period, not a warm-up period, simulation will have an one and half years of warm-up period. The duration of the stable period simulation is three months (90 days).

5.4 The Other Scheduling Models Used For Comparison

Three other scheduling policies will be created to compare with the Concurrent Scheduling model.

a) The Weighted First Come First Serve (WFCFS) model, the current most common scheduling policy in hospital industry. In this model every DRG will have its own weight value. Weight values are assigned with respect to disease severity, charge, and so on. Experienced administrative staffs can assign appropriate weight value which can satisfy the expectation most patients and the medical concerns of hospital staffs.

In this model, when resources are available,

(1) Gather all patients whose weight values are largest.

(2) Choose the earliest arriving patient.

(3) Allocate this resource to him.

b) The Weighted Shortest Process Time (WSPT) model is a very robust classic dispatch model. Like model(a), every DRG has its own weight value. Every patient in the patient queue will have a priority value, which is equal to "weight value divided by process time". The policy will choose the patient with the largest priority value.

c) The mixed model of WSPT and Balanced Work Load (BWL) method: it is similar to model(b), but the patient allocating policy is "Allocate patients who had not chosen his specific physician to the physician whose daily patient time load is shortest" (not the rotating policy).

To see the performance of these models under different resource demands, besides original patient arrival frequency, there are two additional experiments which employ different patient arrival frequencies.

Experiment (1) uses a lower patient arrival frequency.

Experiment (2) uses the original patient arrival frequency.

Experiment (3) uses a higher patient arrival frequency.

5.5 The Performance Evaluation

The objective of this model is to provide hospital administration an overall planning and patient scheduling approach for Medicare, Medicaid, and PPS markets. In these markets, how to balance costs and reimbursements is a primary concern. Since the reimbursement for each DRG is fixed, costs will decide the ranges of profit margins. Costs can be separated into two types: treatment costs and waiting cost. Treatment costs include all resource costs for treatments. The other type is waiting cost, which is caused by a patient waiting for required resources. In these models, treatment costs are assumed fixed and waiting costs will vary according to penalty cost factors, time penalty factors, and the length of waiting time for different DRG and treatment. So waiting costs will be the index of performance in comparing these four models.

There are two kinds of waiting costs in an inpatient operations. They are the following:

1) additional regular medical costs: when inpatients wait for resources, hospitals have additional regular medical costs like nursing costs, overhead costs, and hospital round cost.

2) penalty costs: physical conditions of inpatients will be worse if they don't get treatment in a timely fashion and hospitals have to spend more to them.

The higher the sum of these two waiting costs, the worse the performance of the scheduling model is judged to be.

5.6 The Result Of Simulation:

In these models, total waiting cost is an important relative performance measure of the four models. In addition, the bed occupancy ratio, which can be determined by bed occupancy divided by total bed capacity, is another important index of performance.

Detailed performance data is given in table 2.

As a result of the simulations, there are some interesting findings.

In experiment (1) which employs low arrival frequencies, the differences across the four models are not obvious; in experiment (2), the differences across these four models are more obvious; in experiment (3), the differences across the four models increase greatly.

In experiment (1), model (a) is the worst one. The rest of models have similar performances.

In experiment (2), Concurrent Scheduling, model(d) is best overall. Model (a) is the worst one. The performance of model (b) is between that of model (a) and that of model (c).

		items of costs			
1		additional regular	Experiment (1)	Experiment (2)	Experiment (3)
	model (a)	medical costs	515	714	2200
	model (b)	(unit.$)	263	460	922
	model (c)		342	259	868
	model (d)		321	207	722
2		penalty costs			
	model (a)	(unit.$)	565	666	3501
	model (b)		529	612	1129
	model (c)		519	494	915
	model (d)		509	469	689
3		total waiting costs			
	model (a)	(item1 +item2)	1080	1380	5701
	model (b)		792	1072	2051
	model (c)		861	753	1783
	model (d)		830	676	1411
4		queue length of			
	model (a)	operating rooms	5.0073	6.0021	19.957
	model (b)	(unit.person)	2.0893	4.2615	8.5247
	model (c)		2.22833	2.495	5.9123
	model (d)		1.9069	2.06244	6.3513
5	model (a)	bed occupancy	14.306	15.375	30.193
	model (b)	(unit.bed)	10.063	13.723	18.805
	model (c)		10.217	11.654	15.841
	model (d)		9.8928	11.046	16.366

model (a): WFCFS model
model (b): WSPT model
model (c): the mixed model of WSPT and BWL
model (d): Concurrent Scheduling model

Experiment (1): Demands below capacity
Experiment (2): Demands close to capacity
Experiment (3): Demands over capacity

additional regular medical costs: costs of nursing and other ancillary services
penality costs: costs because of patient physical condition deterioration

Table 2 Result of simulation

In experiment (3) which employs higher arrival frequencies, model (c) and model (d) still look promising although waiting time increases; on the other hand, model (a), and model (b) degrade significantly. In fact, the performance of model (c) is better than that of model (d) in some areas, such as bed occupancy and the operating room queue length. SPT model is the best among the classic dispatch heuristic method, especially when many jobs pile up in a queue. SPT has a major weakness in that a job that has a long process time and a low weight value could stay in a patient queue and be tremendously delayed in getting required resources when resource utilization levels are high.

Appendices 10-13 are Siman-4 simulation programs for the WFCFS model, the WSPT model, the mixed model of WSPT and BWL, and the Concurrent Scheduling model.

Appendices 14-25 are data files for the above four models which are A (WFCFS), B (WSPT), C(the mixed model of WSPT and BWL) and D (Concurrent Scheduling) under three kinds of patient arrival frequency (demands) which are "1" (lower patient arrival frequency), "2" (regular patient arrival frequency), and "3" (higher patient arrival frequency). The model A1 means that model A (WFCFS) under the first kind of patient arrival demand levels.

Appendices 26-37 are output files for these four models under the three kinds of patient arrival demand levels.

6. Conclusion:

The reason why in experiment (1) there is no obvious difference between these models is because most inpatients don't wait long to get an operating room, the functions of balancing work load of same type resources and priority rules don't have many chances to operate.

An interesting result is that severe disease often takes a longer treatment time and higher costs. And they also are the most profitable DRG. Simply stated, these diseases will require a longer operations and need higher technology equipment than ordinary diseases. In this case, DRG3 patients are the most profitable / unit time cases and the severest cases across the three kinds of DRG patients; so, DRG3 patients have the highest priority to get required resources most of the time. According to Stecke and Solberg (1981), the Longest Process Time (LPT) method is not a good scheduling method compared with SPT method with respect to total flow time and equally important jobs. In this hospital application, DRG3 patients have the longest process time and the highest priority by the Concurrent Scheduling priority rule across the three kinds of DRG patients, because they are the most profitable / unit time patients. But the presence of these patients will make total flow time longer. Among these four models, WFCFS considers all factors, such as the severity of diseases and profitability, but its performance is proven that it is not competitive with the other models. WSPT and the mixed model of WSPT and BWL only take care of profitability and lead to severely ill patients waiting too long. Only Concurrent Scheduling efficiently manages the trade off between profitability and disease severity. Thus, although DRG3 patients make total flow time longer, it is a necessary evil.

In addition, Morton and Pentico (1993) classified scheduling levels into five levels. These levels are categorized by different time horizons. What follow are these scheduling levels:

1) long-range planning: time horizon is 2-5 years.
2) middle-range planning: time horizon is 1-2 years.
3) short-range planning: time horizon is 3-6 months.
4) scheduling: time horizon is 2-6 weeks.
5) reactive scheduling / control: time horizon is 1-3 days.

Higher scheduling levels will consider production in a longer term view and lower levels will help higher levels' goals come true, particularly when some unexpected things happen. Basically, a good capacity planning performs excellently when all conditions vary little from what is expected. But in a stochastic environment, anything has the possibility of happening. When demands become surging, lower levels, such as scheduling and reactive scheduling / control will help capacity planning manage these accidents.

By way of simulation experiments, Concurrent Scheduling works better than the rest of the models when patient arrival frequency (demands) become abruptly high. To sum up, the Concurrent Scheduling model is better than the others, especially when real demands are higher than expected demands.

In my simplified inpatient operations example, there are only 36 beds, 3 surgeons, and 1 operation room. The cost differences between WFCFS and Concurrent Scheduling method under different arrival frequencies are given in table 3:

The inserted idleness method is commonly used in manufacturing. In Morton and Pentico (1993), when an important job is coming soon, the inserted idleness method does not let a long and low priority job hold the resource and let the important job wait too long. The inserted idleness method will temporarily idle resources until the important job arrives. In inpatient operations, this method can facilitate emergency patients and high profit margin patients getting required resources. This will cut the average waiting time of important patients and reduce costly waiting costs. Conceptually, this method looks promising.

		items of costs	Experiment (1)	Experiment (2)	Experiment (3)
1		additional regular medical costs			
	WFCFS	(unit.$)	515	714	2200
	Concurrent Scheduling		321	207	722
2		penalty costs			
	WFCFS	(unit.$)	565	666	3501
	Concurrent Scheduling		509	469	689
3		total waiting costs			
	WFCFS	(item1 +item2)	1080	1380	5701
	Concurrent Scheduling		830	676	1411

Experiment (1): Demands below capacity
Experiment (2): Demands close to capacity
Experiment (3): Demands over capacity

additional regular medical costs: costs of nursing and other ancillary services
penality costs: costs because of patient physical condition deterioration

Table 3 The waiting cost differences between WFCFS and Concurrent Scheduling

But, according to the Statistical Reference Index (1993) for hospitalization, patient care hours data for surgeons breaks out in the following way:

1) The mean number of hours in patient direct care per week for surgeons is 49.3. 46% of the hours are spent in office visits; 32% of the hours are spent in hospital rounds; and 13% of the hours are spent in surgery.

2) The mean number of surgical procedures per week for surgery is 8.8 hrs.

3) The mean time devoted to professional activities per week for each surgeon is 59.1 hrs.

The above data suggests that physicians spent most of their time waiting for or involved in office visits or hospital rounds. If the office visit process time or hospital round process time for a single patient is long, such as 30 mins, the inserted idleness method could very possible save significant waiting costs for important patients.

The Concurrent Scheduling model is a Myopic Dispatch method, because it assigns resources to a current highest priority value patient.

Morton and Pentico (1993) propose the following hypotheses:

1) The myopic policy performs better than other simple

dispatch policies,

2) The iterated myopic policy performs better than the myopic policy, and

3) Bottleneck dynamics performs better than the iterated myopic policy.

The myopic policy makes the best current choice without worrying about the future and will get a local optimal solution which is not necessarily a global optimal solution to the patient scheduling problem.

Bottleneck Dynamics first estimates prices (delay costs) for delaying activities and prices (delay costs) for delaying resources. Then the activity which has highest benefit/cost ratio is chosen, because this activity can minimize the delay costs of other resources or activities.

If these hypotheses are correct and general in their applicability, then there is a lot of opportunity for Concurrent Scheduling to improve by applying these extended myopic dispatch policy ideas and is worthy of further study in the future.

			emg.	reg.	surg.	icu	picu	outlier	recovery	death
Pij	transition probability	**emg.**	0	0	0.98	0	0	0	0	0.02
		reg.	0	0	1	0	0	0	0	0
		surg.	0	0	0	0.08	0	0.01	0.9	0.01
		icu	0	0	0	0	0.9	0.05	0	0.05
		picu	0	0	0	0	0	0	1	0
Etk(1)	first moment of process time	**emg.**	0	0	18	0	0	0	0	18
		reg.	0	0	0	0	0	0	0	0
		surg.	0	0	0	440	0	440	440	440
		icu	0	0	0	0	0	0	0	0
		picu	0	0	0	0	0	0	40	0
Etk(2)	second moment of process time	**emg.**	0	0	421	0	0	0	0	421
		reg.	0	0	0	0	0	0	0	0
		surg.	0	0	0	202500	0	202500	202500	202500
		icu	0	0	0	0	0	0	0	0
		picu	0	0	0	0	0	0	2070	0
Mk'		**emg.**	0	0	17.64	0	0	0	0	0.36
		reg.	0	0	0	0	0	0	0	0
		surg.	0	0	0	35.2	0	4.4	396	4.4
		icu	0	0	0	0	0	0	0	0
		picu	0	0	0	0	0	0	40	0
Mk(1)		**emg.**	18	0	0	0	0			
		reg.	0	0	0	0	0			
		surg.	0	0	440	0	0			
		icu	0	0	0	0	0			
		picu	0	0	0	0	40			

Appendix 1 **An example of Semi-Markov model for demands on surgeon per DRG2 patient; (unit.min)**

Pij * Tijk

	emg.	reg.	surg.	icu	picu	outlier	recovery	death
emg.	0	0	412.58	0	0	0	0	8.42
reg.	0	0	0	0	0	0	0	0
surg.	0	0	0	16200	0	2025	182250	2025
icu	0	0	0	0	0	0	0	0
picu	0	0	0	0	0	0	2070	0

Sum(Pij * Tijk)

emg.	421
reg.	0
surg.	202500
icu	0
picu	2070

Mk(2)

emg.	421	0	0	0	0
reg.	0	0	0	0	0
surg.	0	0	202500	0	0
icu	0	0	0	0	0
picu	0	0	0	0	2070

Rk(2)

emg.	421
reg.	0
surg.	202500
icu	0
picu	2070

I

emg.	1	0	0	0	0
reg.	0	1	0	0	0
surg.	0	0	1	0	0
icu	0	0	0	1	0
picu	0	0	0	0	1

Appendix 1 **continued**

		emg.	reg.	surg.	icu	picu
I-Q	emg.	1	0	-0.98	0	0
	reg.	0	1	-1	0	0
	surg.	0	0	1	-0.08	0
	icu	0	0	0	1	-0.9
	picu	0	0	0	0	1
(I-Q)^-1	emg.	1	0	0.98	0.0784	0.07056
	reg.	0	1	1	0.08	0.072
	surg.	0	0	1	0.08	0.072
	icu	0	0	0	1	0.9
	picu	0	0	0	0	1
Nk(1)	emg.	18	0	431.2	0	2.8224
	reg.	0	0	440	0	2.88
	surg.	0	0	440	0	2.88
	icu	0	0	0	0	36
	picu	0	0	0	0	40
Mk' Nk(1)	emg.	0	0	7761.6	0	50.8032
	reg.	0	0	0	0	0
	surg.	0	0	0	0	1267.2
	icu	0	0	0	0	0
	picu	0	0	0	0	0
I # Mk' Nk(1)	emg.	0	0	0	0	0
	reg.	0	0	0	0	0
(#: means the congruent matrix multiplication)	surg.	0	0	0	0	0
	icu	0	0	0	0	0
	picu	0	0	0	0	0

Appendix 1 **continued**

Mk(2)+2I # Mk' Nk(1)

	emg.	reg.	surg.	icu	picu
emg.	421	0	0	0	0
reg.	0	0	0	0	0
surg.	0	0	202500	0	0
icu	0	0	0	0	0
picu	0	0	0	0	2070

Nk(2)

	emg.	reg.	surg.	icu	picu
emg.	421	0	198450	0	146.0592
reg.	0	0	202500	0	149.04
surg.	0	0	202500	0	149.04
icu	0	0	0	0	1863
picu	0	0	0	0	2070

Nk(1)#Nk(1)

	emg.	reg.	surg.	icu	picu
emg.	324	0	185933.4	0	7.965942
reg.	0	0	193600	0	8.2944
surg.	0	0	193600	0	8.2944
icu	0	0	0	0	1296
picu	0	0	0	0	1600

Nk(sd)*Nk(sd)

	emg.	reg.	surg.	icu	picu
emg.	97	0	12516.56	0	138.0933
reg.	0	0	8900	0	140.7456
surg.	0	0	8900	0	140.7456
icu	0	0	0	0	567
picu	0	0	0	0	470

Nk(sd)

	emg.	reg.	surg.	icu	picu
emg.	9.848858	0	111.8774	0	11.75131
reg.	0	0	94.33981	0	11.86363
surg.	0	0	94.33981	0	11.86363
icu	0	0	0	0	23.81176
picu	0	0	0	0	21.67948

Appendix 1 **continued**

column matrix of 1	**emg.**	1
	reg.	1
	surg.	1
	icu	1
	picu	1
		time
V(1)	**emg.**	452.0224
	reg.	442.88
	surg.	442.88
	icu	36
	picu	40
Mk' Vk(1)	**emg.**	7812.403
	reg.	0
	surg.	1267.2
	icu	0
	picu	0
Rk(2)+2Mk'Vk(1)	**emg.**	16045.81
	reg.	0
	surg.	205034.4
	icu	0
	picu	2070
V(2)	**emg.**	217125.6
	reg.	205183.4
	surg.	205183.4
	icu	1863
	picu	2070

Appendix 1 **continued**

		time
Vk(sd) * Vk(sd)	**emg.**	12801.33
	reg.	9040.746
	surg.	9040.746
	icu	567
	picu	470
Vk(sd)	**emg.**	113.143
	reg.	95.08284
	surg.	95.08284
	icu	23.81176
	picu	21.67948
Fk	**emg.**	0.1
	reg.	0.9
	surg.	0
	icu	0
	picu	0
ETtik		96.88885
Monthly arival frequency		20
ETik		1937.777

emg: Emergency admission
reg: Regular admission
surg: Surgery unit
icu: ICU
picu: Post ICU
outlier: outlier (absorbing state)
recovery: recovery (absorbing state)
death: death (absorbing state)

Appendix 1 **continued**

(unit.min)			(length*possibility)
DRG 1	length	possibility	average time
emergency admission		5%	
X-ray	10	1	10
Chem. lab	30	1	30
MRI	30	0.5	15
Elect. lab	30	0.2	6
bed for emg	4320	1	4320
surgeon	200	0.05	10
regular admission		95%	
X-ray	10	1	10
Chem. lab	30	1	30
MRI	30	0.5	15
Elect. lab	30	0.2	6
surgery unit		100%	
bed	5220	1	5220
surgeon	180+20*N	1	260
operating room	180	1	180
X-ray	10	0.2	2
Chem. lab	30	0.4	12
MRI	30	0.2	6
Elect. lab	30	0.02	0.6
radiology lab	120	0.08	9.6
patient room	4320	1	4320
recovery room	720	1	720
ICU		5%	
bed	4320	1	4320
post-ICU		100%	
bed	2880	1	2880
surgeon	20*(2880/1440)	1	40
X-ray	10	0.2	2
Chem. lab	30	0.4	12
MRI	30	0.2	6
Elect. lab	30	0.02	0.6

length: means average process time per treatment or service

possibility: means the possibility that a DRG1 patient will take this service or treatment

average time: means the average process time per DRG1 patient

5%: means 5% of patient will enter this treatment unit

Appendix 2 **Average process time in each treatment unit per DRG1 patient**

(unit.min)			(length*possibility)
DRG 2	length	possibility	average time
emergency admission		8%	
X-ray	10	1	10
Chem. lab	30	1	30
MRI	30	1	30
Elect. lab	30	0.2	6
bed for emg	5760	1	5760
surgeon	360	0.05	18
M.D.	xxxx		
regular admission		92%	
X-ray	10	1	10
Chem. lab	30	1	30
MRI	30	1	30
Elect. lab	30	0.2	6
surgery unit		100%	
bed	6840	1	6840
surgeon	360+20(6840/1440)	1	440
operating room	360	1	360
X-ray	10	0.5	5
Chem. lab	30	1	30
MRI	30	0.5	15
Elect. lab	30	0.1	3
radiology lab	120	0.08	9.6
patient room	5760	1	5760
recovery room	720	1	720
ICU		8%	
bed	5760	1	5760
post-ICU		100%	
bed	4320	1	4320
surgeon	20*(4320/1440)	1	60
X-ray	10	0.5	5
Chem. lab	30	1	30
MRI	30	0.5	15
Elect. lab	30	0.1	3

length: means average process time per treatment or service
possibility: means the possibility that a DRG2 patient will take this service or treatment
average time: means the average process time per DRG1 patient
8%: means 8% of patient will enter this treatment unit

Appendix 3 Average process time in each treatment unit per DRG2 patient

(unit.min)			(length*possibility)
DRG 3	length	possibility	average time
emergency admission		12%	
X-ray	10	1	10
Chem. lab	30	1	30
MRI	30	1	30
Elect. lab	30	1	30
bed for emg	7200	1	7200
surgeon	420	0.05	21
Pre-diagnosis		88%	
X-ray	10	1	10
Chem. lab	30	1	30
MRI	30	1	30
Elect. lab	30	1	30
Admission			
bed	14400	1	14400
surgeon	600+20(14400/1440	1	800
operating room	600	1	600
X-ray	10	0.8	8
Chem. lab	30	1	30
MRI	30	0.8	24
Elect. lab	30	0.8	24
radiology lab	120	0.1	12
patient room	14400	1	14400
recovery room	960	1	960
ICU		10%	
bed	8640	1	8640
post-ICU		100%	
bed	7200	1	7200
surgeon	20*(7200/1440)	1	100
X-ray	10	0.8	8
Chem. lab	30	1	30
MRI	30	0.8	24
Elect. lab	30	0.8	24

length: means average process time per treatment or service
possibility: means the possibility that a DRG3 patient will take this service or treatment
average time: means the average process time per DRG1 patient
12%: means 12% of patient will enter this treatment unit

Appendix 4 Average process time in each treatment unit per DRG3 patient

DRG 1	Pij		emg.	reg.	surg.	icu	picu	outlier	recovery	death
		emg.	0	0	0.99	0	0	0	0	0.01
		reg.	0	0	1	0	0	0	0	0
		surg.	0	0	0	0.05	0	0.01	0.93	0.01
		icu	0	0	0	0	0.9	0.05	0	0.05
		picu	0	0	0	0	0	0	1	0
		outlier	0	0	0	0	0	1	0	0
		recovery	0	0	0	0	0	0	1	0
		death	0	0	0	0	0	0	0	1

DRG 2	Pij		emg.	reg.	surg.	icu	picu	outlier	recovery	death
		emg.	0	0	0.98	0	0	0	0	0.02
		reg.	0	0	1	0	0	0	0	0
		surg.	0	0	0	0.08	0	0.01	0.9	0.01
		icu	0	0	0	0	0.9	0.05	0	0.05
		picu	0	0	0	0	0	0	1	0
		outlier	0	0	0	0	0	1	0	0
		recovery	0	0	0	0	0	0	1	0
		death	0	0	0	0	0	0	0	1

DRG 3	Pij		emg.	reg.	surg.	icu	picu	outlier	recovery	death
		emg.	0	0	0.95	0	0	0	0	0.05
		reg.	0	0	1	0	0	0	0	0
		surg.	0	0	0	0.1	0	0.02	0.86	0.02
		icu	0	0	0	0	0.88	0.06	0	0.06
		picu	0	0	0	0	0	0	1	0
		outlier	0	0	0	0	0	1	0	0
		recovery	0	0	0	0	0	0	1	0
		death	0	0	0	0	0	0	0	1

emg: Emergency admission
reg: Regular admission
surg: Surgery Unit
icu: Intensive care unit
picu: Post intensive care unit
outlier: outlier (absorbing state)
recovery: recovery (absorbing state)
death: death (aborbing state)

Appendix 5 **The transition probability of each DRG**

(unit.day)
Regular patient bed per patient

Average	DRG 1	DRG 2	DRG 3
	3.285	5.286	12.04

Standard Deviation	DRG 1	DRG 2	DRG 3
	2.988	5.351	7.742

Emergency patient bed

Average	DRG 1	DRG 2	DRG 3
	3	**4**	**5**

Standard Deviation	DRG 1	DRG 2	DRG 3
	5.657	**6.083**	**7.416**

regular patient bed: means average stay except emergency room per patient
emergency patient bed: means average emergency bed stay for an emergency admitted patient

Appendix 6 The demand on patient bed per regular admitted patient and emergency admitted patient

	Average	5%	8%	12%
	Resource name	DRG 1	DRG 2	DRG 3
1	surgeon	269.182	452.022	789.36
2	MRI	21.207	45.758	54.806
3	Chem. lab	42.415	61.517	61.008
4	Elect. lab	6.621	9.152	54.806
5	X-ray	12.069	15.253	18.269
6	radiology lab	9.504	9.408	11.4
7	operating room	178.2	352.8	570

	Standard Deviation			
	Resource name	DRG 1	DRG 2	DRG 3
1	surgeon	64.92	113.143	332.463
2	MRI	4.593	20.676	22.928
3	Chem. lab	19.05	25.528	26.418
4	Elect. lab	3.844	3.987	24.015
5	X-ray	6.535	7.567	5.819
6	radiology lab	8.251	7.962	9.46
7	operating room	62.96	96.448	222.261

5% : means "5% of particular DRG type patients are emergency admitted"
Average: means "average montly demand for a particular type DRG"
Standard deviation:
means "the standard deviation of monthly demand for a particular type DR

Appendix 7 **The demands on each kind of resource per emgency admitted patient**

	Average	95%	92%	88%
	Resource name	DRG 1	DRG 2	DRG 3
1	surgeon	250.51	442.88	808.8
2	MRI	21.227	46.08	56.112
3	Chem. lab	42.453	62.16	62.64
4	Elect. lab	6.623	9.216	56.112
5	X-ray	12.076	15.36	18.704
6	radiology lab	9.6	9.6	12
7	operating room	180	360	601

	Standard Deviation			
	Resource name	DRG 1	DRG 2	DRG 3
1	surgeon	76.327	95.083	288.892
2	MRI	4.588	20.377	22.072
3	Chem. lab	19.034	25.189	25.615
4	Elect. lab	3.286	3.427	23.318
5	X-ray	6.24	7.448	5.411
6	radiology lab	8.237	7.927	9.327
7	operating room	60.663	83.066	181.105

95%: means "95% of particular DRG type patients are regular admitted"
Average: means "average montly demand for a particular type DRG"
Standard deviation:
means "the standard deviation of monthly demand for a particular type DR

Appendix 8 **The demands on each kind of resource per regular admitted patient**

	Average	DRG 1	DRG 2	DRG 3
	Resource name			
1	surgeon	251.4436	443.6114	806.4672
2	MRI	21.226	46.05424	55.95528
3	Chem. lab	42.4511	62.10856	62.44416
4	Elect. lab	6.6229	9.21088	55.95528
5	X-ray	12.07565	15.35144	18.6518
6	radiology lab	9.5952	9.58464	11.928
7	operating room	179.91	359.424	597.28

	Standard Deviation			
	Resource name			
1	surgeon	242.1174	417.1739	749.5867
2	MRI	20.39435	44.02398	51.99201
3	Chem. lab	41.28105	59.18212	58.12102
4	Elect. lab	6.483955	8.79297	52.12245
5	X-ray	11.79862	14.72868	17.11186
6	radiology lab	9.52799	9.454829	11.63184
7	operating room	174.0625	338.3859	552.2777

Average: means "average montly demand for a particular type DRG"
Standard deviation:
means "the standard deviation of monthly demand for a particular type DR
Example: average surgeon demands

251.4436 = 95%*250.51 + 5%*269.182

Appendix 9 **The demands on each kind of resource per patient**

```
BEGIN;
            CREATE,3;
            ASSIGN:NUMBER=NUMBER+1:
                 DRG_NO=NUMBER;
            BRANCH,1:IF,DRG_NO<=3,NEW;
NEW         BRANCH,2:ALWAYS,NEXT_S:
                 ALWAYS,W_NEW;
W_NEW       DELAY:EX(DRG_NO,DRG_NO):NEXT(NEW);
NEXT_S      BRANCH,1:WITH,CO(DRG_NO+3),EMG:
                 WITH,(1-CO(DRG_NO+3))*0.7,SUGC:
                 ELSE,SUGNC;
EMG         QUEUE,EMG_BED_Q,0,SCRAP;
            SEIZE:EMG_BED;
            ASSIGN:NO_OF_URGENCE=NO_OF_URGENCE+1:
                 ADMIT_NO=ADMIT_NO+1:
                 PAT_NO=ADMIT_NO:
                 ARRTIME=TNOW:
                 NO_DRG(DRG_NO)=NO_DRG(DRG_NO)+1;
            ROUTE:0,6;
SUGC        ASSIGN:DOCT_NO=DP(DRG_NO+6):
                 NEXT(N_STAT);
SUGNC       ASSIGN:NO_OF_NC=NO_OF_NC+1;
            BRANCH,1:IF,NO_OF_NC==3,RECY:
                 ELSE,REG;
REG         ASSIGN:DOCT_NO=NO_OF_NC:
                 NEXT(N_STAT);
RECY        ASSIGN:DOCT_NO=3:
                 NO_OF_NC=0:
                 NEXT(N_STAT);
N_STAT      ROUTE:0,1;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 1 : FIRST VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT1       STATION,FIRST_VISIT;
            ASSIGN:STAT_NO=1:
                 PROCESSTIME=CO(DRG_NO+9);
GET_OD1     QUEUE,OFFICE_DOCT_Q;
            SEIZE,6:DOCTOR(DOCT_NO);
            DELAY:PROCESSTIME;
            RELEASE:DOCTOR(DOCT_NO);
BAC_1       ROUTE:0,2;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 2  DIAGNOSIS INSPECTIONS
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            STATION,INSPECTION;
            BRANCH,1:IF,AMOD(TNOW,1440)>=540,NDAY:
                 ELSE,TDAY;
```

Appendix 10 computer program for WFCFS model

```
NDAY        DELAY:1441-TNOW;
TDAY        DELAY:300;
            ROUTE:0,3;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 3 FOR POST-DIAGNOSIS VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT6       STATION,FOLLOW_VISIT_STAT;
            ASSIGN:STAT_NO=3:
                 PROCESSTIME=CO(DRG_NO+12);
GET_OD6     QUEUE,SAME_DOCT_Q1;
            SEIZE,6:DOCTOR(DOCT_NO);
            DELAY:PROCESSTIME;
            RELEASE:DOCTOR(DOCT_NO);
BAC_6       ROUTE:0,4;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 4 FOR ADMISSION
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT7       STATION,ADMISSION_STAT;
            QUEUE,ADMISSION_Q;
            SEIZE:BED;
            ASSIGN:ADMIT_NO=ADMIT_NO+1:
                 PAT_NO=ADMIT_NO:
                 NO_DRG(DRG_NO)=NO_DRG(DRG_NO)+1:
                 IN_HOSP(PAT_NO)=1;
            DUPLICATE:1,HOSP;
            ROUTE:0,5;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 5 FOR SUGERY UNIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT8       STATION,SURGERY_STAT;
            ASSIGN:STAT_NO=5:
                 PROCESSTIME=RN(DRG_NO+15):
                 WEIGHT=CO(DRG_NO+18):
                 LATE_COST=CO(DRG_NO+33):
                 WAIT_DUE_LENGTH=CO(DRG_NO+36):
                 PROFIT_MARGIN=CO(DRG_NO+39):
                 MED_COST=CO(DRG_NO+47):
                 ARRTIME=TNOW;
            BRANCH,1:IF,DAY>5,W_OPER:
                 IF,AMOD(TNOW,1440)<240,GET_S:

IF,AMOD(TNOW,1440)>302.AND.AMOD(TNOW,1440)<540
                         .AND.REST(DOCT_NO)==0,GET_S:
                 ELSE,W_OPER;
GET_S       QUEUE,OPERATING_ROOM_Q,0,W_OPER;
            SEIZE:OPERATING_ROOM;
DOCT_Q2     QUEUE,SAME_DOCT_Q2;
```

Appendix 10 continued

```
              SEIZE:DOCTOR(DOCT_NO):
                   NEXT(DEL8);
DEL8          BRANCH,1:IF,DRG_NO==1,ST51:
                   IF,DRG_NO==2,ST52:
                   IF,DRG_NO==3,ST53;
ST51          TALLY:WAITING_TIME_FOR_DRG1,TNOW-ARRTIME;
              TALLY:AVERAGE_DRG1_WAITING_TIME,LATE_COST*
                   EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
              ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
                   EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
                   WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
                   MED_COST/1440:
                   NEXT(ST54);
ST52          TALLY:WAITING_TIME_FOR_DRG2,TNOW-ARRTIME;
              TALLY:AVERAGE_DRG2_WAITING_TIME,LATE_COST*
                   EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
              ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
                   EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
                   WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
                   MED_COST/1440:
                   NEXT(ST54);
ST53          TALLY:WAITING_TIME_FOR_DRG3,TNOW-ARRTIME;
              TALLY:AVERAGE_DRG3_WAITING_TIME,LATE_COST*
                   EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
              ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
                   EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
                   WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
                   MED_COST/1440:
                   NEXT(ST54);
ST54          DELAY:PROCESSTIME;                  OPERATING ROOM
              BRANCH,1:IF,AMOD(TNOW,1440)>=240.AND.
                   NQ(DOCT_NO+11).NE.0.AND.DAY<6,RREL:
                   IF,AMOD(TNOW,1440)>=240.AND.NQ(15).NE.0.AND.
                   DAY>5,RREL:
                   IF,AMOD(TNOW,1440)>=540,RREL:
                   ELSE,DUP8;
RREL          RELEASE:OPERATING_ROOM;
              RELEASE:DOCTOR(DOCT_NO):NEXT(RDEL);
DUP8          DUPLICATE:1,FINDB8;
RDEL          BRANCH,1:IF,NOT_ORG==0,ST8_6;
ST8_6         DELAY:CO(DRG_NO+21);                 RECOVERY ROOM
              RELEASE:BED;
              ASSIGN:IN_HOSP(PAT_NO)=0;
              BRANCH,1:IF,DRG_NO==1,COUNT1:
                   IF,DRG_NO==2,COUNT2:
                   IF,DRG_NO==3,COUNT3;
COUNT1             COUNT:SURGERY_OF_DRG1:DISPOSE;
COUNT2             COUNT:SURGERY_OF_DRG2:DISPOSE;
COUNT3             COUNT:SURGERY_OF_DRG3:DISPOSE;
```

Appendix 10 continued

```
FINDB8     ASSIGN:NOT_ORG=1;
           BRANCH,1:IF,DAY>5,ST8_1:
                ELSE,ST8_2;
ST8_1      BRANCH,1:IF,NQ((8+WEEKEND_SURG_NO))==0,RREL:
                ELSE,S8REL;
S8         RELEASE:OPERATING_ROOM;
           DUPLICATE:1,REDOC:NEXT(GET_S);
S8REL      SEARCH,(8+WEEKEND_SURG_NO),1,NQ:
                MAX(WEIGHT);
ST8_4      REMOVE:J,(8+WEEKEND_SURG_NO),ST8_3:DISPOSE;
ST8_3      ASSIGN:CURR_PAT_NO=PAT_NO;
           BRANCH,1:IF,DOCT_NO==WEEKEND_SURG_NO,S8:
                ELSE,ST8_5;
ST8_5      DUPLICATE:1,ST8_1;
           DELAY:1:NEXT(BK_OP);
ST8_2      RELEASE:OPERATING_ROOM;
           DUPLICATE:1,OPCYCLE;
REDOC      DELAY:1;
           RELEASE:DOCTOR(DOCT_NO):DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 6 FOR EMERGENCE ADMISSION
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT10     STATION,EMERGENCY_ROOM_STAT;
           ASSIGN:STAT_NO=6:
                PROCESSTIME=RN(DRG_NO+24):
                WEIGHT=CO(DRG_NO+27);
           BRANCH,1:WITH,0.05,SEMG:
                ELSE,REMG;
SEMG       BRANCH,1:IF,EMG==0,OEMG:
                ELSE,NEMG;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;      11111111111111111111111111
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OEMG       BRANCH,1:IF,DAY>5,WEMG:
                IF,AMOD(TNOW,1440)>540,OTEMG:
                ELSE,RTEMG;
WEMG       FINDJ,1,3:NR(DOCTOR(J))==0;
           ASSIGN:EMG=J;
           BRANCH,1:IF,EMG==WEEKEND_SURG_NO,
                EA1:
                ELSE,EA2;
EA1        BRANCH,1:IF,EMG==3,EA6:
                ELSE,EA7;
EA6        ASSIGN:EMG=1:NEXT(EA2);
EA7        ASSIGN:EMG=WEEKEND_SURG_NO+1:NEXT(EA2);
EA2        ASSIGN:BEGIN_AT_WEEKEND=1;
           BRANCH,2:ALWAYS,WEMG1:
                ALWAYS,WEMG2;
```

Appendix 10 continued

```
WEMG1       QUEUE,EMERGENCY_ROOM_Q3;
            SEIZE,2:DOCTOR(EMG):
                 NEXT(DEL10);
WEMG2       ALTER:DOCTOR(EMG),+1:
                 DISPOSE;


RTEMG       QUEUE,EMERGENCY_ROOM_Q1;
            SELECT,RAN:DOCT1:
                 DOCT2:
                 DOCT3;
DOCT1       SEIZE,2:DOCTOR(1);
            ASSIGN:DOCT_NO=1:
                 EMG=DOCT_NO:
                 NEXT(IFOFF);
DOCT2       SEIZE,2:DOCTOR(2);
            ASSIGN:DOCT_NO=2:
                 EMG=DOCT_NO:
                 NEXT(IFOFF);
DOCT3       SEIZE,2:DOCTOR(3);
            ASSIGN:DOCT_NO=3:
                 EMG=DOCT_NO:
                 NEXT(IFOFF);
IFOFF       BRANCH,1:IF,NQ((EMG+15))==0,IFLUN:
                 ELSE,REMO;
REMO        REMOVE:1,15+EMG,SCRAP:NEXT(IFOFF);
IFLUN       BRANCH,1:IF,NQ((EMG+11))==0,DEL10:
                 ELSE,REML;
REML        REMOVE:1,EMG+11,SS5:NEXT(IFLUN);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OTEMG       BRANCH,1:IF,NR(DOCTOR(1))==1.AND.NR(DOCTOR(2))==1
                 .AND.NR(DOCTOR(3))==1,ROFF:
                 ELSE,OFFD;
ROFF        ASSIGN:OFF_TIME=1:NEXT(RTEMG);
OFFD        ASSIGN:BEGIN_AT_OFF=1;
            FINDJ,1,3:NR(DOCTOR(J))==0;
            ASSIGN:EMG=J;
            BRANCH,2:ALWAYS,OFFQ:
                 ALWAYS,ALTD;
OFFQ        ASSIGN:OFF_TIME=1;
            QUEUE,EMERGENCY_ROOM_Q2;
            SEIZE,2:DOCTOR(EMG):
                 NEXT(DEL10);
ALTD        ALTER:DOCTOR(EMG),+1:
                 DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 10 continued

```
WEMG1       QUEUE,EMERGENCY_ROOM_Q3;
            SEIZE,2:DOCTOR(EMG):
                 NEXT(DEL10);
WEMG2       ALTER:DOCTOR(EMG),+1:
                 DISPOSE;


RTEMG       QUEUE,EMERGENCY_ROOM_Q1;
            SELECT,RAN:DOCT1:
                 DOCT2:
                 DOCT3;
DOCT1       SEIZE,2:DOCTOR(1);
            ASSIGN:DOCT_NO=1:
                 EMG=DOCT_NO:
                 NEXT(IFOFF);
DOCT2       SEIZE,2:DOCTOR(2);
            ASSIGN:DOCT_NO=2:
                 EMG=DOCT_NO:
                 NEXT(IFOFF);
DOCT3       SEIZE,2:DOCTOR(3);
            ASSIGN:DOCT_NO=3:
                 EMG=DOCT_NO:
                 NEXT(IFOFF);
IFOFF       BRANCH,1:IF,NQ((EMG+15))==0,IFLUN:
                 ELSE,REMO;
REMO        REMOVE:1,15+EMG,SCRAP:NEXT(IFOFF);
IFLUN       BRANCH,1:IF,NQ((EMG+11))==0,DEL10:
                 ELSE,REML;
REML        REMOVE:1,EMG+11,SS5:NEXT(IFLUN);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OTEMG       BRANCH,1:IF,NR(DOCTOR(1))==1.AND.NR(DOCTOR(2))==1
                 .AND.NR(DOCTOR(3))==1,ROFF:
                 ELSE,OFFD;
ROFF        ASSIGN:OFF_TIME=1:NEXT(RTEMG);
OFFD        ASSIGN:BEGIN_AT_OFF=1;
            FINDJ,1,3:NR(DOCTOR(J))==0;
            ASSIGN:EMG=J;
            BRANCH,2:ALWAYS,OFFQ:
                 ALWAYS,ALTD;
OFFQ        ASSIGN:OFF_TIME=1;
            QUEUE,EMERGENCY_ROOM_Q2;
            SEIZE,2:DOCTOR(EMG):
                 NEXT(DEL10);
ALTD        ALTER:DOCTOR(EMG),+1:
                 DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 10 continued

```
DEL10       DELAY:PROCESSTIME;
            ASSIGN:DOCT_NO=EMG:
                 EMG=0:NEXT(MERG1);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;       2222222222222222222222222222222222
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
NEMG        BRANCH,1:IF,DAY>5,NWEMG:
                 IF,AMOD(TNOW,1440)>540,OTEMG1:
                 ELSE,RTEMG1;
NWEMG       FINDJ,1,3:NR(DOCTOR(J))==0;
            ASSIGN:EMG1=J;
            BRANCH,1:IF,EMG1==WEEKEND_SURG_NO,
                 EA11:
                 ELSE,EA21;
EA11        BRANCH,1:IF,EMG1==3,EA61:
                 ELSE,EA71;
EA61        ASSIGN:EMG1=1:NEXT(EA21);
EA71        ASSIGN:EMG1=WEEKEND_SURG_NO+1:NEXT(EA21);
EA21        ASSIGN:BEGIN_AT_WEEKEND=1;
            BRANCH,2:ALWAYS,WEMG11:
                 ALWAYS,WEMG21;
WEMG11      QUEUE,EMERGENCY_ROOM_Q6;
            SEIZE,2:DOCTOR(EMG1):
                 NEXT(DEL101);
WEMG21      ALTER:DOCTOR(EMG1),+1:
                 DISPOSE;


RTEMG1      QUEUE,EMERGENCY_ROOM_Q4;
            SELECT,RAN:DOCT11:
                 DOCT21:
                 DOCT31;
DOCT11      SEIZE,2:DOCTOR(1);
            ASSIGN:DOCT_NO=1:
                 EMG1=DOCT_NO:
                 NEXT(IFOFF1);
DOCT21      SEIZE,2:DOCTOR(2);
            ASSIGN:DOCT_NO=2:
                 EMG1=DOCT_NO:
                 NEXT(IFOFF1);
DOCT31      SEIZE,2:DOCTOR(3);
            ASSIGN:DOCT_NO=3:
                 EMG1=DOCT_NO:
                 NEXT(IFOFF1);
IFOFF1      BRANCH,1:IF,NQ((EMG1+15))==0,IFLUN1:
                 ELSE,REMO1;
REMO1       REMOVE:1,15+EMG1,SCRAP:NEXT(IFOFF1);
```

Appendix 10 continued

```
IFLUN1      BRANCH,1:IF,NQ((EMG1+11))==0,DEL101:
                 ELSE,REML1;
REML1       REMOVE:1,EMG1+11,SS5:NEXT(IFLUN1);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OTEMG1      BRANCH,1:IF,NR(DOCTOR(1))==1.AND.NR(DOCTOR(2))==1
                 .AND.NR(DOCTOR(3))==1,ROFF1:
                 ELSE,OFFD1;
ROFF1       ASSIGN:OFF_TIME=1:NEXT(RTEMG1);
OFFD1       ASSIGN:BEGIN_AT_OFF=1;
            FINDJ,1,3:NR(DOCTOR(J))==0;
            ASSIGN:EMG1=J;
            BRANCH,2:ALWAYS,OFFQ1:
                 ALWAYS,ALTD1;
OFFQ1       ASSIGN:OFF_TIME=1;
            QUEUE,EMERGENCY_ROOM_Q5;
            SEIZE,2:DOCTOR(EMG1):
                 NEXT(DEL101);
ALTD1       ALTER:DOCTOR(EMG1),+1:
                 DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

DEL101      DELAY:PROCESSTIME;
            ASSIGN:DOCT_NO=EMG1:
                 EMG1=0:NEXT(MERG1);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

MERG1       BRANCH,1:IF,AMOD(TNOW,1440)<=240,ST10_1:
                 IF,AMOD(TNOW,1440)<=300,ST10_2:
                 IF,AMOD(TNOW,1440)<=540,ST10_1:
                 IF,AMOD(TNOW,1440)>540,ST10_3;
ST10_1      BRANCH,1:IF,BEGIN_AT_WEEKEND==0.AND.DAY>5,EA4:
                 IF,BEGIN_AT_WEEKEND>0.AND.DAY>5,EA4:
                 IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
                 AMOD(TNOW,1440)<480,ST10_2:
                 IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
                 AMOD(TNOW,1440)>=480,ST10_3:
                 ELSE,EA5;
EA4         ALTER:DOCTOR(DOCT_NO),-1;
EA5         RELEASE:DOCTOR(DOCT_NO):NEXT(STABT);
ST10_2      ALTER:DOCTOR(DOCT_NO),-1;
            RELEASE:DOCTOR(DOCT_NO);
            DUPLICATE:1,SCH1:NEXT(STABT);
ST10_3      ALTER:DOCTOR(DOCT_NO),-1;
            RELEASE:DOCTOR(DOCT_NO);
            DUPLICATE:1,SCH2:NEXT(STABT);
SCH1        BRANCH,1:IF,BEGIN_AT_WEEKEND==0.AND.DAY<6,EA3:
```

Appendix 10 continued

```
                IF,BEGIN_AT_WEEKEND==1.AND.DAY==1,EA3;
EA3         DELAY:60;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            BRANCH,1:IF,BEGIN_AT_WEEKEND==1.AND.DAY==1,MON:
                ELSE,NMON;
MON         DUPLICATE:1,SS5;
NMON        DUPLICATE:1,OPCYCLE;
            DELAY:1;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            ALTER:DOCTOR(DOCT_NO),+1:
                DISPOSE;
SCH2        BRANCH,1:IF,BEGIN_AT_OFF.NE.1.AND.DAY<5.AND.
                BEGIN_AT_WEEKEND==0,
                ST10_4:
                IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
                    AMOD(TNOW,1440)>=480,ST10_4;
ST10_4      DELAY:1439-AMOD(TNOW,1440);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            DUPLICATE:1,OPCYCLE;
            DELAY:2;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            ALTER:DOCTOR(DOCT_NO),+1:
                DISPOSE;
STABT       COUNT:SPE_EMG;
            ASSIGN:NO_OF_SPE_EMG=NO_OF_SPE_EMG+1;
            DELAY:RN(DRG_NO+30)-PROCESSTIME:
                NEXT(PROCT);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
REMG            DELAY:RN(DRG_NO+30);
PROCT       ASSIGN:NO_OF_EMGPAT=NO_OF_EMGPAT+1;
            BRANCH,1:IF,NO_OF_EMGPAT==3,RECY1:
                ELSE,REGCY;
REGCY       ASSIGN:DOCT_NO=NO_OF_EMGPAT:
                NEXT(N_STAT1);
RECY1       ASSIGN:DOCT_NO=3:
                NO_OF_EMGPAT=0:
                NEXT(N_STAT1);
N_STAT1     RELEASE:EMG_BED;
            ROUTE:0,4;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; THE TOTAL NO. OF EACH DRG
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            CREATE;
            DELAY:655200;
            ASSIGN:NO_DRG(1)=0:
                NO_DRG(2)=0:
                NO_DRG(3)=0:
```

Appendix 10 continued

```
          DELAYED_COST=0:
          WAIT_COST=0;
     DELAY:129600;
     CREATE;
     DELAY:784800;
     COUNT:TOTAL_DRG1,NO_DRG(1);
     COUNT:TOTAL_DRG2,NO_DRG(2);
     COUNT:TOTAL_DRG3,NO_DRG(3);
     COUNT:TOTAL_DELAYED_COST,DELAYED_COST;
     COUNT:TOTAL_WAIT_COST,WAIT_COST:
          DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; QUEUE FOR OPERATING ROOM

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
W_OPER    ASSIGN:W_OPER_TIME=TNOW:
               2ND_TRY=1:
               BEG=0;
BK_OP     QUEUE,22:DETACH;
OPCYCLE   BRANCH,1:IF,NQ(22)>0,OPSEAR;
OPSEAR    SEARCH,(22),1,NQ:
               MAX(WEIGHT);
OR13      REMOVE:J,(22),OR5:DISPOSE;
OR5       ASSIGN:CURR_PAT_NO=PAT_NO;
          BRANCH,1:IF,DOCT_NO==EMG,OR1:
               IF,DOCT_NO==EMG1,OR1:
               ELSE,GET_S;
OR1       DUPLICATE:1,OPCYCLE;
          DELAY:1:NEXT(BK_OP);


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; SCHEDULE FOR SURGEONS
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          CREATE:1440;
          DELAY:1;
          ASSIGN:TIME_SCHEDULE=1;
          BRANCH,3:ALWAYS,SURG1:
               ALWAYS,SURG2:
               ALWAYS,SURG3;
SURG1     ASSIGN:DOCT_NO=1:NEXT(SSCH);
SURG2     ASSIGN:DOCT_NO=2:NEXT(SSCH);
SURG3     ASSIGN:DOCT_NO=3:NEXT(SSCH);
SSCH      ASSIGN:BEG=1;
          BRANCH,1:IF,DAY<=5,WDAY1:
               IF,DAY==7,WEND1;
WEND1     DELAY:1438:NEXT(REG1);
WDAY1     DELAY:240;
```

Appendix 10 continued

```
            BRANCH,1:IF,EMG==DOCT_NO,NOLU1:
                 IF,EMG1==DOCT_NO,NOLU1:
                 ELSE,LU1;
LU1         QUEUE,(DOCT_NO+11);
            SEIZE,3:DOCTOR(DOCT_NO);
            ALTER:DOCTOR(DOCT_NO),-1;
            RELEASE:DOCTOR(DOCT_NO);
            BRANCH,1:IF,AMOD(TNOW,1440)>=480,ODEL1:
                 ELSE,LDEL1;
LDEL1       ASSIGN:REST(DOCT_NO)=1;
            DELAY:60;
            BRANCH,1:IF,NR(OPERATING_ROOM).NE.1,SS1:
                 ELSE,SS2;
SS1         DUPLICATE:1,OPCYCLE;
SS2         DELAY:1;
            ALTER:DOCTOR(DOCT_NO),+1;
            ASSIGN:REST(DOCT_NO)=0;
SS5         DELAY:542-AMOD(TNOW,1440);
OBRA1       BRANCH,1:IF,EMG.NE.DOCT_NO,OFF1:
                 IF,EMG1.NE.DOCT_NO,OFF1;
OFF1        QUEUE,(15+DOCT_NO);
            SEIZE,3:DOCTOR(DOCT_NO);
            ALTER:DOCTOR(DOCT_NO),-1;
            RELEASE:DOCTOR(DOCT_NO);
ODEL1       ASSIGN:REST(DOCT_NO)=1;
            DELAY:1439-AMOD(TNOW,1440);
            BRANCH,1:IF,DAY.NE.5,REG1;
REG1        DUPLICATE:1,OPCYCLE;
SS3         BRANCH,1:IF,DOCT_NO.NE.EMG.AND.
                 DOCT_NO.NE.EMG1,SS4;
SS4         DELAY:2;
            ASSIGN:REST(DOCT_NO)=0;
            ALTER:DOCTOR(DOCT_NO),+1:
                 DISPOSE;
NOLU1       DELAY:301:NEXT(OBRA1);

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; SCHEDULE FOR WEEKEND SURGEON
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            CREATE:10080;
            ASSIGN:TIME_SCHEDULE=1;
            DELAY:7199;
WFIND       ASSIGN:WEEKEND_SURG_NO=LAST_SURG;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
WS4         BRANCH,1:IF,WEEKEND_SURG_NO==EMG,WS5:
                 IF,WEEKEND_SURG_NO==EMG1,WS5:
                 ELSE,WS6;
WS5         BRANCH,1:IF,WEEKEND_SURG_NO==3,WS1:
                 ELSE,WS2;
```

Appendix 10 continued

```
WS1         ASSIGN:WEEKEND_SURG_NO=1:
                 LAST_SURG=WEEKEND_SURG_NO+1:
                 NEXT(WS4);
WS2         ASSIGN:WEEKEND_SURG_NO=WEEKEND_SURG_NO+1:
                 NEXT(WS4);
WS6         BRANCH,1:IF,WEEKEND_SURG_NO==3,WS7:
                 ELSE,WS8;
WS7         ASSIGN:LAST_SURG=1:
                 NEXT(SAMES);
WS8         ASSIGN:LAST_SURG=WEEKEND_SURG_NO+1:
                 NEXT(SAMES);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SAMES       BRANCH,1:IF,NQ(22)>0,WSEAR3:
                 ELSE,NWS1;
WSEAR3      SEARCH,(22),1,NQ:
                 MAX(WEIGHT);
            REMOVE:J,(22),FOR_OP:DISPOSE;
NWS1        DELAY:2;
            ALTER:DOCTOR(WEEKEND_SURG_NO),+1;
            DELAY:240;
            QUEUE,WEEKEND_LUNCH_SURG_Q;
            SEIZE:DOCTOR(WEEKEND_SURG_NO);
            ALTER:DOCTOR(WEEKEND_SURG_NO),-1;
            RELEASE:DOCTOR(WEEKEND_SURG_NO);
            BRANCH,1:IF,AMOD(TNOW,1440)>=480,WSODEL:
                 ELSE,WSLDEL;
WSLDEL      DELAY:60;
            BRANCH,1:IF,NQ((22))>0,WSEAR2:
                 ELSE,NWS2;
WSEAR2      SEARCH,(22),1,NQ:
                 MAX(WEIGHT);
            REMOVE:J,(22),FOR_OP:DISPOSE;
NWS2        DELAY:1;
            ALTER:DOCTOR(WEEKEND_SURG_NO),+1;
            DELAY:542-AMOD(TNOW,1440);
            QUEUE,WEEKEND_OFF_SURG_Q;
            SEIZE:DOCTOR(WEEKEND_SURG_NO);
            ALTER:DOCTOR(WEEKEND_SURG_NO),-1;
            RELEASE:DOCTOR(WEEKEND_SURG_NO);
WSODEL      DELAY:1439-AMOD(TNOW,1440);
            BRANCH,1:IF,DAY==6,WFIND;


FOR_OP      BRANCH,1:IF,DOCT_NO==WEEKEND_SURG_NO,GET_WS:
                 ELSE,NWS3;
NWS3        DUPLICATE:1,SAMES;
            DELAY:1:NEXT(BK_OP);
GET_WS      BRANCH,1:IF,AMOD(TNOW,1440)==1439,WDU1:
                 ELSE,WDU2;
```

Appendix 10 continued

```
WDU1        DUPLICATE:1,NWS1:NEXT(WQ);
WDU2        DUPLICATE:1,NWS2:NEXT(WQ);
WQ          QUEUE,WEEKEND_OPERATING_Q;
            SEIZE,2:OPERATING_ROOM;
            QUEUE,WEEKEND_SURG_Q;
            SEIZE,2:DOCTOR(WEEKEND_SURG_NO):
                 NEXT(DEL8);

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;      WEEKDAY CALENDER
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            CREATE:1440;
NEWD        ASSIGN:DAY=DAY+1;
            BRANCH,1:IF,DAY==7,NWEEK;
NWEEK       DELAY:1439;
            ASSIGN:DAY=0:
                 WEEK=WEEK+1:
                 DISPOSE;

SCRAP       COUNT:NO_USE:DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;      OVER_NIGHT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            CREATE:1440;
            ASSIGN:CLD=CLD+1;
            BRANCH,1:IF,CLD==1,DD8:
                 IF,CLD==7,DD8:
                 ELSE,DD7;
DD7         ASSIGN:CL=CL+1;
            BRANCH,1:IF,CL<4,DD1:
            ELSE,DD2;
DD1         BRANCH,1:IF,NQ((CL+11))==0,DD4:
                 ELSE,DD3;
DD3         REMOVE:1,(CL+11),SCRAP:NEXT(DD1);
DD4         BRANCH,1:IF,NQ((CL+15))==0,DD7:
                 ELSE,DD6;
DD6         REMOVE:1,(CL+15),SCRAP:NEXT(DD4);
DD2         ASSIGN:CL=0:DISPOSE;

DD8         BRANCH,1:IF,NQ(15)==0,DD9:
                 ELSE,DD10;
DD10        REMOVE:1,15,SCRAP:NEXT(DD8);
DD9         BRANCH,1:IF,NQ(19)==0,DD13:
                 ELSE,DD11;
DD11        REMOVE:1,19,SCRAP:NEXT(DD9);
DD13        BRANCH,1:IF,CLD==7,DD12;
DD12        ASSIGN:CLD=0:DISPOSE;
```

Appendix 10 continued

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;     OFFICE VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
           CREATE;
OUTPAT     BRANCH,2:ALWAYS,NEXTO:
                ALWAYS,NEWO;
NEWO       DELAY:EX(47):NEXT(OUTPAT);
NEXTO      QUEUE,OUT_PAT_Q;
           SELECT,POR:OUT1:
                OUT2:
                OUT3;
OUT1       SEIZE,7:DOCTOR(1);
           ASSIGN:DOCT_NO=1:NEXT(OUT4);
OUT2       SEIZE,7:DOCTOR(2);
           ASSIGN:DOCT_NO=2:NEXT(OUT4);
OUT3       SEIZE,7:DOCTOR(3);
           ASSIGN:DOCT_NO=3:NEXT(OUT4);
OUT4       DELAY:CO(43);
           RELEASE:DOCTOR(DOCT_NO):DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;     HOSPITAL RUN
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
HOSP       DELAY:1442-AMOD(TNOW,1440);
           BRANCH,1:IF,IN_HOSP(PAT_NO)==1.AND.DAY<6,SEE;
SEE        QUEUE,HOSP_RUN_Q;
           SEIZE,5:DOCTOR(DOCT_NO);
           DELAY:CO(43+DRG_NO);
           RELEASE:DOCTOR(DOCT_NO):NEXT(HOSP);

END;
```

Appendix 10 continued

```
BEGIN;
            CREATE,3;
            ASSIGN:NUMBER=NUMBER+1:
                 DRG_NO=NUMBER;
            BRANCH,1:IF,DRG_NO<=3,NEW;
NEW         BRANCH,2:ALWAYS,NEXT_S:
                 ALWAYS,W_NEW;
W_NEW       DELAY:EX(DRG_NO,DRG_NO):NEXT(NEW);
NEXT_S      BRANCH,1:WITH,CO(DRG_NO+3),EMG:
                 WITH,(1-CO(DRG_NO+3))*0.7,SUGC:
                 ELSE,SUGNC;
EMG         QUEUE,EMG_BED_Q,0,SCRAP;
            SEIZE:EMG_BED;
            ASSIGN:NO_OF_URGENCE=NO_OF_URGENCE+1:
                 ADMIT_NO=ADMIT_NO+1:
                 PAT_NO=ADMIT_NO:
                 ARRTIME=TNOW:
                 NO_DRG(DRG_NO)=NO_DRG(DRG_NO)+1;
            ROUTE:0,6;
SUGC        ASSIGN:DOCT_NO=DP(DRG_NO+6):
                 NEXT(N_STAT);
SUGNC       ASSIGN:NO_OF_NC=NO_OF_NC+1;
            BRANCH,1:IF,NO_OF_NC==3,RECY:
                 ELSE,REG;
REG         ASSIGN:DOCT_NO=NO_OF_NC:
                 NEXT(N_STAT);
RECY        ASSIGN:DOCT_NO=3:
                 NO_OF_NC=0:
                 NEXT(N_STAT);
N_STAT      ROUTE:0,1;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 1 : FIRST VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT1       STATION,FIRST_VISIT;
            ASSIGN:STAT_NO=1:
                 PROCESSTIME=CO(DRG_NO+9);
GET_OD1     QUEUE,OFFICE_DOCT_Q;
            SEIZE,6:DOCTOR(DOCT_NO);
            DELAY:PROCESSTIME;
            RELEASE:DOCTOR(DOCT_NO);
BAC_1       ROUTE:0,2;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 2  DIAGNOSIS INSPECTIONS
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            STATION,INSPECTION;
            BRANCH,1:IF,AMOD(TNOW,1440)>=540,NDAY:
                 ELSE,TDAY;
```

Appendix 11 computer program for WSPT model

```
NDAY        DELAY:1441-TNOW;
TDAY        DELAY:300;
            ROUTE:0,3;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 3 FOR POST-DIAGNOSIS VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT6       STATION,FOLLOW_VISIT_STAT;
            ASSIGN:STAT_NO=3:
                 PROCESSTIME=CO(DRG_NO+12);
GET_OD6     QUEUE,SAME_DOCT_Q1;
            SEIZE,6:DOCTOR(DOCT_NO);
            DELAY:PROCESSTIME;
            RELEASE:DOCTOR(DOCT_NO);
BAC_6       ROUTE:0,4;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 4 FOR ADMISSION
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT7       STATION,ADMISSION_STAT;
            QUEUE,ADMISSION_Q;
            SEIZE:BED;
            ASSIGN:ADMIT_NO=ADMIT_NO+1:
                 PAT_NO=ADMIT_NO:
                 NO_DRG(DRG_NO)=NO_DRG(DRG_NO)+1:
                 IN_HOSP(PAT_NO)=1;
            DUPLICATE:1,HOSP;
            ROUTE:0,5;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 5 FOR SUGERY UNIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT8       STATION,SURGERY_STAT;
            ASSIGN:STAT_NO=5:
                 PROCESSTIME=RN(DRG_NO+15):
                 WEIGHT=CO(DRG_NO+18):
                 LATE_COST=CO(DRG_NO+33):
                 WAIT_DUE_LENGTH=CO(DRG_NO+36):
                 PROFIT_MARGIN=CO(DRG_NO+39):
                 MED_COST=CO(DRG_NO+47):
                 ARRTIME=TNOW;
            BRANCH,1:IF,DAY>5,W_OPER:
                 IF,AMOD(TNOW,1440)<240,GET_S:

IF,AMOD(TNOW,1440)>302.AND.AMOD(TNOW,1440)<540
                      .AND.REST(DOCT_NO)==0,GET_S:
                 ELSE,W_OPER;
GET_S       QUEUE,OPERATING_ROOM_Q,0,W_OPER;
            SEIZE:OPERATING_ROOM;
DOCT_Q2     QUEUE,SAME_DOCT_Q2;
```

Appendix 11 continued

```
                SEIZE:DOCTOR(DOCT_NO):
                     NEXT(DEL8);
DEL8            BRANCH,1:IF,DRG_NO==1,ST51:
                     IF,DRG_NO==2,ST52:
                     IF,DRG_NO==3,ST53;
ST51            TALLY:WAITING_TIME_FOR_DRG1,TNOW-ARRTIME;
                TALLY:AVERAGE_DRG1_WAITING_TIME,LATE_COST*
                     EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
                ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
                     EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
                     WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
                     MED_COST/1440:
                     NEXT(ST54);
ST52            TALLY:WAITING_TIME_FOR_DRG2,TNOW-ARRTIME;
                TALLY:AVERAGE_DRG2_WAITING_TIME,LATE_COST*
                     EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
                ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
                     EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
                     WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
                     MED_COST/1440:
                     NEXT(ST54);
ST53            TALLY:WAITING_TIME_FOR_DRG3,TNOW-ARRTIME;
                TALLY:AVERAGE_DRG3_WAITING_TIME,LATE_COST*
                     EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
                ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
                     EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
                     WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
                     MED_COST/1440:
                     NEXT(ST54);
ST54            DELAY:PROCESSTIME;                  OPERATING ROOM
                BRANCH,1:IF,AMOD(TNOW,1440)>=240.AND.
                     NQ(DOCT_NO+11).NE.0.AND.DAY<6,RREL:
                     IF,AMOD(TNOW,1440)>=240.AND.NQ(15).NE.0.AND.
                     DAY>5,RREL:
                     IF,AMOD(TNOW,1440)>=540,RREL:
                     ELSE,DUP8;
RREL            RELEASE:OPERATING_ROOM;
                RELEASE:DOCTOR(DOCT_NO):NEXT(RDEL);
DUP8            DUPLICATE:1,FINDB8;
RDEL            BRANCH,1:IF,NOT_ORG==0,ST8_6;
ST8_6           DELAY:CO(DRG_NO+21);                 RECOVERY ROOM
                RELEASE:BED;
                ASSIGN:IN_HOSP(PAT_NO)=0;
                BRANCH,1:IF,DRG_NO==1,COUNT1:
                     IF,DRG_NO==2,COUNT2:
                     IF,DRG_NO==3,COUNT3;
COUNT1               COUNT:SURGERY_OF_DRG1:DISPOSE;
COUNT2               COUNT:SURGERY_OF_DRG2:DISPOSE;
COUNT3               COUNT:SURGERY_OF_DRG3:DISPOSE;
```

Appendix 11 continued

```
FINDB8      ASSIGN:NOT_ORG=1;
            BRANCH,1:IF,DAY>5,ST8_1:
                 ELSE,ST8_2;
ST8_1       BRANCH,1:IF,NQ((8+WEEKEND_SURG_NO))==0,RREL:

                 ELSE,S8REL;
S8          RELEASE:OPERATING_ROOM;
            DUPLICATE:1,REDOC:NEXT(GET_S);
S8REL       SEARCH,(8+WEEKEND_SURG_NO),1,NQ:
                 MAX(WEIGHT/PROCESSTIME);
ST8_4       REMOVE:J,(8+WEEKEND_SURG_NO),ST8_3:DISPOSE;
ST8_3       ASSIGN:CURR_PAT_NO=PAT_NO;
            BRANCH,1:IF,DOCT_NO==WEEKEND_SURG_NO,S8:
                 ELSE,ST8_5;
ST8_5       DUPLICATE:1,ST8_1;
            DELAY:1:NEXT(BK_OP);
ST8_2       RELEASE:OPERATING_ROOM;
            DUPLICATE:1,OPCYCLE;
REDOC       DELAY:1;
            RELEASE:DOCTOR(DOCT_NO):DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 6 FOR EMERGENCE ADMISSION
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT10      STATION,EMERGENCY_ROOM_STAT;
            ASSIGN:STAT_NO=6:
                 PROCESSTIME=RN(DRG_NO+24):
                 WEIGHT=CO(DRG_NO+27);
            BRANCH,1:WITH,0.05,SEMG:
                 ELSE,REMG;
SEMG        BRANCH,1:IF,EMG==0,OEMG:
                 ELSE,NEMG;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;       1111111111111111111111111
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OEMG        BRANCH,1:IF,DAY>5,WEMG:
                 IF,AMOD(TNOW,1440)>540,OTEMG:
                 ELSE,RTEMG;
WEMG        FINDJ,1,3:NR(DOCTOR(J))==0;
            ASSIGN:EMG=J;
            BRANCH,1:IF,EMG==WEEKEND_SURG_NO,
                 EA1:
                 ELSE,EA2;
EA1         BRANCH,1:IF,EMG==3,EA6:
                 ELSE,EA7;
EA6         ASSIGN:EMG=1:NEXT(EA2);
EA7         ASSIGN:EMG=WEEKEND_SURG_NO+1:NEXT(EA2);
EA2         ASSIGN:BEGIN_AT_WEEKEND=1;
            BRANCH,2:ALWAYS,WEMG1:
```

Appendix 11 continued

```
                ALWAYS,WEMG2;
WEMG1       QUEUE,EMERGENCY_ROOM_Q3;
            SEIZE,2:DOCTOR(EMG):
                NEXT(DEL10);
WEMG2       ALTER:DOCTOR(EMG),+1:
                DISPOSE;


RTEMG       QUEUE,EMERGENCY_ROOM_Q1;
            SELECT,RAN:DOCT1:
                DOCT2:
                DOCT3;
DOCT1       SEIZE,2:DOCTOR(1);
            ASSIGN:DOCT_NO=1:
                EMG=DOCT_NO:
                NEXT(IFOFF);
DOCT2       SEIZE,2:DOCTOR(2);
            ASSIGN:DOCT_NO=2:
                EMG=DOCT_NO:
                NEXT(IFOFF);
DOCT3       SEIZE,2:DOCTOR(3);
            ASSIGN:DOCT_NO=3:
                EMG=DOCT_NO:
                NEXT(IFOFF);
IFOFF       BRANCH,1:IF,NQ((EMG+15))==0,IFLUN:
                ELSE,REMO;
REMO        REMOVE:1,15+EMG,SCRAP:NEXT(IFOFF);
IFLUN       BRANCH,1:IF,NQ((EMG+11))==0,DEL10:
                ELSE,REML;
REML        REMOVE:1,EMG+11,SS5:NEXT(IFLUN);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OTEMG       BRANCH,1:IF,NR(DOCTOR(1))==1.AND.NR(DOCTOR(2))==1
                .AND.NR(DOCTOR(3))==1,ROFF:
                ELSE,OFFD;
ROFF        ASSIGN:OFF_TIME=1:NEXT(RTEMG);
OFFD        ASSIGN:BEGIN_AT_OFF=1;
            FINDJ,1,3:NR(DOCTOR(J))==0;
            ASSIGN:EMG=J;
            BRANCH,2:ALWAYS,OFFQ:
                ALWAYS,ALTD;
OFFQ        ASSIGN:OFF_TIME=1;
            QUEUE,EMERGENCY_ROOM_Q2;
            SEIZE,2:DOCTOR(EMG):
                NEXT(DEL10);
ALTD        ALTER:DOCTOR(EMG),+1:
                DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 11 continued

```
DEL10        DELAY:PROCESSTIME;
             ASSIGN:DOCT_NO=EMG:
                  EMG=0:NEXT(MERG1);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;        22222222222222222222222222222222
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
NEMG         BRANCH,1:IF,DAY>5,NWEMG:
                  IF,AMOD(TNOW,1440)>540,OTEMG1:
                  ELSE,RTEMG1;
NWEMG        FINDJ,1,3:NR(DOCTOR(J))==0;
             ASSIGN:EMG1=J;
             BRANCH,1:IF,EMG1==WEEKEND_SURG_NO,
                  EA11:
                  ELSE,EA21;
EA11         BRANCH,1:IF,EMG1==3,EA61:
                  ELSE,EA71;
EA61         ASSIGN:EMG1=1:NEXT(EA21);
EA71         ASSIGN:EMG1=WEEKEND_SURG_NO+1:NEXT(EA21);
EA21         ASSIGN:BEGIN_AT_WEEKEND=1;
             BRANCH,2:ALWAYS,WEMG11:
                  ALWAYS,WEMG21;
WEMG11       QUEUE,EMERGENCY_ROOM_Q6;
             SEIZE,2:DOCTOR(EMG1):
                  NEXT(DEL101);
WEMG21       ALTER:DOCTOR(EMG1),+1:
                  DISPOSE;

RTEMG1       QUEUE,EMERGENCY_ROOM_Q4;
             SELECT,RAN:DOCT11:
                  DOCT21:
                  DOCT31;
DOCT11       SEIZE,2:DOCTOR(1);
             ASSIGN:DOCT_NO=1:
                  EMG1=DOCT_NO:
                  NEXT(IFOFF1);
DOCT21       SEIZE,2:DOCTOR(2);
             ASSIGN:DOCT_NO=2:
                  EMG1=DOCT_NO:
                  NEXT(IFOFF1);
DOCT31       SEIZE,2:DOCTOR(3);
             ASSIGN:DOCT_NO=3:
                  EMG1=DOCT_NO:
                  NEXT(IFOFF1);
IFOFF1       BRANCH,1:IF,NQ((EMG1+15))==0,IFLUN1:
                  ELSE,REMO1;
REMO1        REMOVE:1,15+EMG1,SCRAP:NEXT(IFOFF1);
IFLUN1       BRANCH,1:IF,NQ((EMG1+11))==0,DEL101:
                  ELSE,REML1;
```

Appendix 11 continued

```
REML1     REMOVE:1,EMG1+11,SS5:NEXT(IFLUN1);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OTEMG1    BRANCH,1:IF,NR(DOCTOR(1))==1.AND.NR(DOCTOR(2))==1
               .AND.NR(DOCTOR(3))==1,ROFF1:
               ELSE,OFFD1;
ROFF1     ASSIGN:OFF_TIME=1:NEXT(RTEMG1);
OFFD1     ASSIGN:BEGIN_AT_OFF=1;
          FINDJ,1,3:NR(DOCTOR(J))==0;
          ASSIGN:EMG1=J;
          BRANCH,2:ALWAYS,OFFQ1:
               ALWAYS,ALTD1;
OFFQ1     ASSIGN:OFF_TIME=1;
          QUEUE,EMERGENCY_ROOM_Q5;
          SEIZE,2:DOCTOR(EMG1):
               NEXT(DEL101);
ALTD1     ALTER:DOCTOR(EMG1),+1:
               DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

DEL101    DELAY:PROCESSTIME;
          ASSIGN:DOCT_NO=EMG1:
               EMG1=0:NEXT(MERG1);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

MERG1     BRANCH,1:IF,AMOD(TNOW,1440)<=240,ST10_1:
               IF,AMOD(TNOW,1440)<=300,ST10_2:
               IF,AMOD(TNOW,1440)<=540,ST10_1:
               IF,AMOD(TNOW,1440)>540,ST10_3;
ST10_1    BRANCH,1:IF,BEGIN_AT_WEEKEND==0.AND.DAY>5,EA4:
               IF,BEGIN_AT_WEEKEND>0.AND.DAY>5,EA4:
               IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
               AMOD(TNOW,1440)<480,ST10_2:
               IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
               AMOD(TNOW,1440)>=480,ST10_3:
               ELSE,EA5;
EA4       ALTER:DOCTOR(DOCT_NO),-1;
EA5       RELEASE:DOCTOR(DOCT_NO):NEXT(STABT);
ST10_2    ALTER:DOCTOR(DOCT_NO),-1;
          RELEASE:DOCTOR(DOCT_NO);
          DUPLICATE:1,SCH1:NEXT(STABT);
ST10_3    ALTER:DOCTOR(DOCT_NO),-1;
          RELEASE:DOCTOR(DOCT_NO);
          DUPLICATE:1,SCH2:NEXT(STABT);
SCH1      BRANCH,1:IF,BEGIN_AT_WEEKEND==0.AND.DAY<6,EA3:
               IF,BEGIN_AT_WEEKEND==1.AND.DAY==1,EA3;
EA3       DELAY:60;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 11 continued

```
          BRANCH,1:IF,BEGIN_AT_WEEKEND==1.AND.DAY==1,MON:
               ELSE,NMON;
MON       DUPLICATE:1,SS5;
NMON      DUPLICATE:1,OPCYCLE;
          DELAY:1;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          ALTER:DOCTOR(DOCT_NO),+1:
               DISPOSE;
SCH2      BRANCH,1:IF,BEGIN_AT_OFF.NE.1.AND.DAY<5.AND.
               BEGIN_AT_WEEKEND==0,
               ST10_4:
               IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
                    AMOD(TNOW,1440)>=480,ST10_4;
ST10_4    DELAY:1439-AMOD(TNOW,1440);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          DUPLICATE:1,OPCYCLE;
          DELAY:2;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          ALTER:DOCTOR(DOCT_NO),+1:
               DISPOSE;
STABT     COUNT:SPE_EMG;
          ASSIGN:NO_OF_SPE_EMG=NO_OF_SPE_EMG+1;
          DELAY:RN(DRG_NO+30)-PROCESSTIME:
               NEXT(PROCT);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
REMG           DELAY:RN(DRG_NO+30);
PROCT     ASSIGN:NO_OF_EMGPAT=NO_OF_EMGPAT+1;
          BRANCH,1:IF,NO_OF_EMGPAT==3,RECY1:
               ELSE,REGCY;
REGCY     ASSIGN:DOCT_NO=NO_OF_EMGPAT:
               NEXT(N_STAT1);
RECY1     ASSIGN:DOCT_NO=3:
               NO_OF_EMGPAT=0:
               NEXT(N_STAT1);
N_STAT1   RELEASE:EMG_BED;
          ROUTE:0,4;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; THE TOTAL NO. OF EACH DRG
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          CREATE;
          DELAY:655200;
          ASSIGN:NO_DRG(1)=0:
               NO_DRG(2)=0:
               NO_DRG(3)=0:
               DELAYED_COST=0:
               WAIT_COST=0;
          DELAY:129600;
          CREATE;
```

Appendix 11 continued

```
          DELAY:784800;
          COUNT:TOTAL_DRG1,NO_DRG(1);
          COUNT:TOTAL_DRG2,NO_DRG(2);
          COUNT:TOTAL_DRG3,NO_DRG(3);
          COUNT:TOTAL_DELAYED_COST,DELAYED_COST;
          COUNT:TOTAL_WAIT_COST,WAIT_COST:
               DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; QUEUE FOR OPERATING ROOM
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
W_OPER    ASSIGN:W_OPER_TIME=TNOW:
               2ND_TRY=1:
               BEG=0;
BK_OP     QUEUE,22:DETACH;
OPCYCLE   BRANCH,1:IF,NQ(22)>0,OPSEAR;
OPSEAR    SEARCH,(22),1,NQ:
               MAX(WEIGHT/PROCESSTIME);
OR13      REMOVE:J,(22),OR5:DISPOSE;
OR5       ASSIGN:CURR_PAT_NO=PAT_NO;
          BRANCH,1:IF,DOCT_NO==EMG,OR1:
               IF,DOCT_NO==EMG1,OR1:
               ELSE,GET_S;
OR1       DUPLICATE:1,OPCYCLE;
          DELAY:1:NEXT(BK_OP);


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; SCHEDULE FOR SURGEONS
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          CREATE:1440;
          DELAY:1;
          ASSIGN:TIME_SCHEDULE=1;
          BRANCH,3:ALWAYS,SURG1:
               ALWAYS,SURG2:
               ALWAYS,SURG3;
SURG1     ASSIGN:DOCT_NO=1:NEXT(SSCH);
SURG2     ASSIGN:DOCT_NO=2:NEXT(SSCH);
SURG3     ASSIGN:DOCT_NO=3:NEXT(SSCH);
SSCH      ASSIGN:BEG=1;
          BRANCH,1:IF,DAY<=5,WDAY1:
               IF,DAY==7,WEND1;
WEND1     DELAY:1438:NEXT(REG1);


WDAY1     DELAY:240;
          BRANCH,1:IF,EMG==DOCT_NO,NOLU1:
               IF,EMG1==DOCT_NO,NOLU1:
               ELSE,LU1;
LU1       QUEUE,(DOCT_NO+11);
```

Appendix 11 continued

```
            SEIZE,3:DOCTOR(DOCT_NO);
            ALTER:DOCTOR(DOCT_NO),-1;
            RELEASE:DOCTOR(DOCT_NO);
            BRANCH,1:IF,AMOD(TNOW,1440)>=480,ODEL1:
                 ELSE,LDEL1;
LDEL1       ASSIGN:REST(DOCT_NO)=1;
            DELAY:60;
            BRANCH,1:IF,NR(OPERATING_ROOM).NE.1,SS1:
                 ELSE,SS2;
SS1         DUPLICATE:1,OPCYCLE;
SS2         DELAY:1;
            ALTER:DOCTOR(DOCT_NO),+1;
            ASSIGN:REST(DOCT_NO)=0;
SS5         DELAY:542-AMOD(TNOW,1440);
OBRA1       BRANCH,1:IF,EMG.NE.DOCT_NO,OFF1:
                 IF,EMG1.NE.DOCT_NO,OFF1;
OFF1        QUEUE,(15+DOCT_NO);
            SEIZE,3:DOCTOR(DOCT_NO);
            ALTER:DOCTOR(DOCT_NO),-1;
            RELEASE:DOCTOR(DOCT_NO);
ODEL1       ASSIGN:REST(DOCT_NO)=1;
            DELAY:1439-AMOD(TNOW,1440);
            BRANCH,1:IF,DAY.NE.5,REG1;
REG1        DUPLICATE:1,OPCYCLE;
SS3         BRANCH,1:IF,DOCT_NO.NE.EMG.AND.
                 DOCT_NO.NE.EMG1,SS4;
SS4         DELAY:2;
            ASSIGN:REST(DOCT_NO)=0;
            ALTER:DOCTOR(DOCT_NO),+1:
                 DISPOSE;
NOLU1       DELAY:301:NEXT(OBRA1);

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; SCHEDULE FOR WEEKEND SURGEON
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            CREATE:10080;
            ASSIGN:TIME_SCHEDULE=1;
            DELAY:7199;
WFIND       ASSIGN:WEEKEND_SURG_NO=LAST_SURG;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
WS4         BRANCH,1:IF,WEEKEND_SURG_NO==EMG,WS5:
                 IF,WEEKEND_SURG_NO==EMG1,WS5:
                 ELSE,WS6;
WS5         BRANCH,1:IF,WEEKEND_SURG_NO==3,WS1:
                 ELSE,WS2;
WS1         ASSIGN:WEEKEND_SURG_NO=1:
                 LAST_SURG=WEEKEND_SURG_NO+1:
                 NEXT(WS4);
WS2         ASSIGN:WEEKEND_SURG_NO=WEEKEND_SURG_NO+1:
                 NEXT(WS4);
```

Appendix 11 continued

```
WS6         BRANCH,1:IF,WEEKEND_SURG_NO==3,WS7:
                 ELSE,WS8;
WS7         ASSIGN:LAST_SURG=1:
                 NEXT(SAMES);
WS8         ASSIGN:LAST_SURG=WEEKEND_SURG_NO+1:
                 NEXT(SAMES);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SAMES       BRANCH,1:IF,NQ(22)>0,WSEAR3:
                 ELSE,NWS1;
WSEAR3      SEARCH,(22),1,NQ:
                 MAX(WEIGHT);
            REMOVE:J,(22),FOR_OP:DISPOSE;
NWS1        DELAY:2;
            ALTER:DOCTOR(WEEKEND_SURG_NO),+1;
            DELAY:240;
            QUEUE,WEEKEND_LUNCH_SURG_Q;
            SEIZE:DOCTOR(WEEKEND_SURG_NO);
            ALTER:DOCTOR(WEEKEND_SURG_NO),-1;
            RELEASE:DOCTOR(WEEKEND_SURG_NO);
            BRANCH,1:IF,AMOD(TNOW,1440)>=480,WSODEL:
                 ELSE,WSLDEL;
WSLDEL      DELAY:60;
            BRANCH,1:IF,NQ((22))>0,WSEAR2:
                 ELSE,NWS2;
WSEAR2      SEARCH,(22),1,NQ:
                 MAX(WEIGHT/PROCESSTIME);
            REMOVE:J,(22),FOR_OP:DISPOSE;
NWS2        DELAY:1;
            ALTER:DOCTOR(WEEKEND_SURG_NO),+1;
            DELAY:542-AMOD(TNOW,1440);
            QUEUE,WEEKEND_OFF_SURG_Q;
            SEIZE:DOCTOR(WEEKEND_SURG_NO);
            ALTER:DOCTOR(WEEKEND_SURG_NO),-1;
            RELEASE:DOCTOR(WEEKEND_SURG_NO);
WSODEL      DELAY:1439-AMOD(TNOW,1440);
            BRANCH,1:IF,DAY==6,WFIND;

FOR_OP      BRANCH,1:IF,DOCT_NO==WEEKEND_SURG_NO,GET_WS:
                 ELSE,NWS3;
NWS3        DUPLICATE:1,SAMES;
            DELAY:1:NEXT(BK_OP);
GET_WS      BRANCH,1:IF,AMOD(TNOW,1440)==1439,WDU1:
                 ELSE,WDU2;
WDU1        DUPLICATE:1,NWS1:NEXT(WQ);
WDU2        DUPLICATE:1,NWS2:NEXT(WQ);
WQ          QUEUE,WEEKEND_OPERATING_Q;
            SEIZE,2:OPERATING_ROOM;
            QUEUE,WEEKEND_SURG_Q;
            SEIZE,2:DOCTOR(WEEKEND_SURG_NO):
                 NEXT(DEL8);
```

Appendix 11 continued

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;    WEEKDAY CALENDER
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          CREATE:1440;
NEWD      ASSIGN:DAY=DAY+1;
          BRANCH,1:IF,DAY==7,NWEEK;
NWEEK     DELAY:1439;
          ASSIGN:DAY=0:
               WEEK=WEEK+1:
               DISPOSE;
SCRAP     COUNT:NO_USE:DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;    OVER_NIGHT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          CREATE:1440;
          ASSIGN:CLD=CLD+1;
          BRANCH,1:IF,CLD==1,DD8:
               IF,CLD==7,DD8:
               ELSE,DD7;
DD7       ASSIGN:CL=CL+1;
          BRANCH,1:IF,CL<4,DD1:
          ELSE,DD2;
DD1       BRANCH,1:IF,NQ((CL+11))==0,DD4:
               ELSE,DD3;
DD3       REMOVE:1,(CL+11),SCRAP:NEXT(DD1);
DD4       BRANCH,1:IF,NQ((CL+15))==0,DD7:
               ELSE,DD6;
DD6       REMOVE:1,(CL+15),SCRAP:NEXT(DD4);
DD2       ASSIGN:CL=0:DISPOSE;
DD8       BRANCH,1:IF,NQ(15)==0,DD9:
               ELSE,DD10;
DD10      REMOVE:1,15,SCRAP:NEXT(DD8);
DD9       BRANCH,1:IF,NQ(19)==0,DD13:
               ELSE,DD11;
DD11      REMOVE:1,19,SCRAP:NEXT(DD9);
DD13      BRANCH,1:IF,CLD==7,DD12;
DD12      ASSIGN:CLD=0:DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;    OFFICE VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          CREATE;
OUTPAT    BRANCH,2:ALWAYS,NEXTO:
               ALWAYS,NEWO;
NEWO      DELAY:EX(47):NEXT(OUTPAT);
NEXTO     QUEUE,OUT_PAT_Q;
          SELECT,POR:OUT1:
               OUT2:
               OUT3;
```

Appendix 11 continued

```
OUT1        SEIZE,7:DOCTOR(1);
            ASSIGN:DOCT_NO=1:NEXT(OUT4);
OUT2        SEIZE,7:DOCTOR(2);
            ASSIGN:DOCT_NO=2:NEXT(OUT4);
OUT3        SEIZE,7:DOCTOR(3);
            ASSIGN:DOCT_NO=3:NEXT(OUT4);
OUT4        DELAY:CO(43);
            RELEASE:DOCTOR(DOCT_NO):DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;      HOSPITAL RUN
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
HOSP        DELAY:1442-AMOD(TNOW,1440);
            BRANCH,1:IF,IN_HOSP(PAT_NO)==1.AND.DAY<6,SEE;
SEE         QUEUE,HOSP_RUN_Q;
            SEIZE,5:DOCTOR(DOCT_NO);
            DELAY:CO(43+DRG_NO);
            RELEASE:DOCTOR(DOCT_NO):NEXT(HOSP);

END;
```

Appendix 11 continued

```
BEGIN;
            CREATE,3;
            ASSIGN:NUMBER=NUMBER+1:
                 DRG_NO=NUMBER;
            BRANCH,1:IF,DRG_NO<=3,NEW;
NEW         BRANCH,2:ALWAYS,NEXT_S:
                 ALWAYS,W_NEW;
W_NEW       DELAY:EX(DRG_NO,DRG_NO):NEXT(NEW);
NEXT_S      BRANCH,1:WITH,CO(DRG_NO+3),EMG:
                 WITH,(1-CO(DRG_NO+3))*0.7,SUGC:
                 ELSE,SUGNC;
EMG         QUEUE,EMG_BED_Q,0,SCRAP;
            SEIZE:EMG_BED;
            ASSIGN:NO_OF_URGENCE=NO_OF_URGENCE+1:
                 ADMIT_NO=ADMIT_NO+1:
                 PAT_NO=ADMIT_NO:
                 ARRTIME=TNOW:
                 NO_DRG(DRG_NO)=NO_DRG(DRG_NO)+1;
            ROUTE:0,6;
SUGC        ASSIGN:DOCT_NO=DP(DRG_NO+6):
                 EXPECT_PAT_PROC_TIME=CO(DRG_NO+9)
                 +CO(DRG_NO+12):
                 EXPECT_PAT_SURG_TIME=CO(DRG_NO+9):
                 DOCT_PROC_TIME(DOCT_NO)=
                 DOCT_PROC_TIME(DOCT_NO)+
                 EXPECT_PAT_PROC_TIME:
                 NEXT(N_STAT);
SUGNC       ASSIGN:EXPECT_PAT_PROC_TIME=CO(DRG_NO+9)+
                 CO(DRG_NO+12):
                 EXPECT_PAT_SURG_TIME=CO(DRG_NO+9);
            FINDJ,1,3:MIN(DOCT_PROC_TIME(J));
            ASSIGN:DOCT_NO=J:
                 DOCT_PROC_TIME(J)=DOCT_PROC_TIME(J)
                 +EXPECT_PAT_PROC_TIME:
                 NEXT(N_STAT);
N_STAT      ROUTE:0,1;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 1 : FIRST VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT1       STATION,FIRST_VISIT;
            ASSIGN:STAT_NO=1:
                 PROCESSTIME=CO(DRG_NO+15);
GET_OD1     QUEUE,OFFICE_DOCT_Q;
            SEIZE,6:DOCTOR(DOCT_NO);
            DELAY:PROCESSTIME;
            RELEASE:DOCTOR(DOCT_NO);
BAC_1       ROUTE:0,2;
```

Appendix 12 computer program for mixed model of WSPT and BWL

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 2  DIAGNOSIS INSPECTIONS
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

          STATION,INSPECTION;
          BRANCH,1:IF,AMOD(TNOW,1440)>=540,NDAY:
               ELSE,TDAY;
NDAY      DELAY:1441-TNOW;
TDAY      DELAY:300;
          ROUTE:0,3;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 3 FOR POST-DIAGNOSIS VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT6     STATION,FOLLOW_VISIT_STAT;
          ASSIGN:STAT_NO=3:
               PROCESSTIME=CO(DRG_NO+18);
GET_OD6   QUEUE,SAME_DOCT_Q1;
          SEIZE,6:DOCTOR(DOCT_NO);
          DELAY:PROCESSTIME;
          RELEASE:DOCTOR(DOCT_NO);
BAC_6     ROUTE:0,4;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 4 FOR ADMISSION
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT7     STATION,ADMISSION_STAT;
          QUEUE,ADMISSION_Q;
          SEIZE:BED;
          ASSIGN:ADMIT_NO=ADMIT_NO+1:
               PAT_NO=ADMIT_NO:
               IN_HOSP(PAT_NO)=1:
               DOCT_SURG_TIME(DOCT_NO)=
               DOCT_SURG_TIME(DOCT_NO)
               +CO(DRG_NO+9):
               NO_DRG(DRG_NO)=NO_DRG(DRG_NO)+1;

          DUPLICATE:1,HOSP;
          ROUTE:0,5;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 5 FOR SUGERY UNIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT8     STATION,SURGERY_STAT;
          ASSIGN:STAT_NO=5:
               PROCESSTIME=RN(DRG_NO+21):
               LATE_COST=CO(DRG_NO+24):
               WAIT_DUE_LENGTH=CO(DRG_NO+27):
               PROFIT_MARGIN=CO(DRG_NO+30):
               MED_COST=CO(DRG_NO+56):
```

Appendix 12 continued

```
                ARRTIME=TNOW;
          BRANCH,1:IF,DAY>5,W_OPER:
                IF,AMOD(TNOW,1440)<240,GET_S:
                IF,AMOD(TNOW,1440)>302.AND.AMOD(TNOW,1440)
                <540.AND.REST(DOCT_NO)==0,GET_S:
                ELSE,W_OPER;
GET_S     QUEUE,OPERATING_ROOM_Q,0,W_OPER;
          SEIZE:OPERATING_ROOM;
DOCT_Q2   QUEUE,SAME_DOCT_Q2;
          SEIZE:DOCTOR(DOCT_NO):
                NEXT(DEL8);
DEL8      BRANCH,1:IF,DRG_NO==1,ST51:
                IF,DRG_NO==2,ST52:
                IF,DRG_NO==3,ST53;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
ST51      TALLY:WAITING_TIME_FOR_DRG1,TNOW-ARRTIME;
          TALLY:AVERAGE_DRG1_WAITING_TIME,LATE_COST*
                EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
          ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
                EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
                WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
                MED_COST/1440:
                NEXT(ST54);
ST52      TALLY:WAITING_TIME_FOR_DRG2,TNOW-ARRTIME;
          TALLY:AVERAGE_DRG2_WAITING_TIME,LATE_COST*
                EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
          ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
                EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
                WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
                MED_COST/1440:
                NEXT(ST54);
ST53      TALLY:WAITING_TIME_FOR_DRG3,TNOW-ARRTIME;
          TALLY:AVERAGE_DRG3_WAITING_TIME,LATE_COST*
                EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
          ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
                EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
                WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
                MED_COST/1440:
                NEXT(ST54);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
ST54      ASSIGN:BEG_SURG_TIME=TNOW;
          ASSIGN:DOCT_PROC_TIME(DOCT_NO)=
                DOCT_PROC_TIME(DOCT_NO)-CO(DRG_NO+9):
                DOCT_SURG_TIME(DOCT_NO)=
                DOCT_SURG_TIME(DOCT_NO)-CO(DRG_NO+9);
          DELAY:PROCESSTIME;                 OPERATING ROOM
                BRANCH,1:IF,AMOD(TNOW,1440)>=240.AND.
                NQ((DOCT_NO+11)).NE.0.AND.DAY<=5,RREL:
                IF,AMOD(TNOW,1440)>=240.AND.NQ(15).NE.0
                .AND.DAY>5,RREL:
```

Appendix 12 continued

```
                 IF,AMOD(TNOW,1440)>=540,RREL:
                 ELSE,DUP8;
RREL        RELEASE:OPERATING_ROOM;
            RELEASE:DOCTOR(DOCT_NO):
                 NEXT(RDEL);
DUP8        DUPLICATE:1,FINDB8;
RDEL        BRANCH,1:IF,NOT_ORG==0,ST8_6;
ST8_6       DELAY:CO(DRG_NO+33);                RECOVERY ROOM
            RELEASE:BED;
            ASSIGN:IN_HOSP(PAT_NO)=0;
            ASSIGN:DOCT_PROC_TIME(DOCT_NO)=
                 DOCT_PROC_TIME(DOCT_NO)
                 -CO(DRG_NO+12):
                 IN_HOSP(PAT_NO)=0;
            BRANCH,1:IF,DRG_NO==1,COUNT1:
                 IF,DRG_NO==2,COUNT2:
                 IF,DRG_NO==3,COUNT3;
COUNT1      COUNT:SURGERY_OF_DRG1:DISPOSE;
COUNT2      COUNT:SURGERY_OF_DRG2:DISPOSE;
COUNT3      COUNT:SURGERY_OF_DRG3:DISPOSE;

FINDB8      ASSIGN:NOT_ORG=1;
            BRANCH,1:IF,DAY>5,ST8_1:
                 ELSE,ST8_2;
ST8_1       BRANCH,1:IF,NQ((8+WEEKEND_SURG_NO))==0,RREL:
                 ELSE,S8;
S8          RELEASE:OPERATING_ROOM;
            DUPLICATE:1,REDOC;
            ASSIGN:CURR=WEEKEND_SURG_NO;
            SEARCH,(8+WEEKEND_SURG_NO),1,NQ:
                 MAX(PROFIT_MARGIN/
                 PROCESSTIME);
            REMOVE:J,(8+WEEKEND_SURG_NO),GET_S:
                 DISPOSE;
ST8_2       RELEASE:OPERATING_ROOM;
            DUPLICATE:1,REDOC;
            ASSIGN:CURR=DOCT_NO:NEXT(OPCYCLE);
REDOC       DELAY:1;
            RELEASE:DOCTOR(DOCT_NO):DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 6 FOR EMERGENCE ADMISSION
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT10      STATION,EMERGENCY_ROOM_STAT;
            ASSIGN:STAT_NO=6:
                 PROCESSTIME=RN(DRG_NO+36):
                 LATE_COST=CO(DRG_NO+39):
                 WAIT_DUE_LENGTH=CO(DRG_NO+42):

                 PROFIT_MARGIN=CO(DRG_NO+45);
```

Appendix 12 continued

```
          BRANCH,1:WITH,0.05,SEMG:
               ELSE,REMG;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SEMG      BRANCH,1:IF,EMG==0,OEMG:
               ELSE,NEMG;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; 11111111111111111
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OEMG      BRANCH,1:IF,DAY>5,WEMG:
               IF,AMOD(TNOW,1440)>540,OTEMG:
               ELSE,RTEMG;
WEMG      FINDJ,1,3:NR(DOCTOR(J))==0;
          ASSIGN:EMG=J;
          BRANCH,1:IF,EMG==WEEKEND_SURG_NO,
               EA1:
               ELSE,EA2;
EA1       ASSIGN:EMG=WEEKEND_SURG_NO+1;
EA2       ASSIGN:BEGIN_AT_WEEKEND=1;
          BRANCH,2:ALWAYS,WEMG1:
               ALWAYS,WEMG2;
WEMG1     QUEUE,EMERGENCY_ROOM_Q3;
          SEIZE,2:DOCTOR(EMG):
               NEXT(DEL10);
WEMG2     ALTER:DOCTOR(EMG),+1:
               DISPOSE;


RTEMG     QUEUE,EMERGENCY_ROOM_Q1;
          SELECT,RAN:DOCT1:
               DOCT2:
               DOCT3;
DOCT1     SEIZE,2:DOCTOR(1);
          ASSIGN:DOCT_NO=1:
               EMG=DOCT_NO:
               NEXT(IFOFF);
DOCT2     SEIZE,2:DOCTOR(2);
          ASSIGN:DOCT_NO=2:
               EMG=DOCT_NO:
               NEXT(IFOFF);
DOCT3     SEIZE,2:DOCTOR(3);
          ASSIGN:DOCT_NO=3:
               EMG=DOCT_NO:
               NEXT(IFOFF);
IFOFF     BRANCH,1:IF,NQ((EMG+15))==0,IFLUN:
               ELSE,REMO;
REMO      REMOVE:1,15+EMG,SCRAP:NEXT(IFOFF);
IFLUN     BRANCH,1:IF,NQ((EMG+11))==0,DEL10:
               ELSE,REML;
```

Appendix 12 continued

```
REML        REMOVE:1,EMG+11,SS5:NEXT(IFLUN);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OTEMG       BRANCH,1:IF,NR(DOCTOR(1))==1.AND.NR(DOCTOR(2))==1
                 .AND.NR(DOCTOR(3))==1,ROFF:
                 ELSE,OFFD;
ROFF        ASSIGN:OFF_TIME=1:NEXT(RTEMG);
OFFD        ASSIGN:BEGIN_AT_OFF=1;
            FINDJ,1,3:NR(DOCTOR(J))==0;
            ASSIGN:EMG=J;
            BRANCH,2:ALWAYS,OFFQ:
                 ALWAYS,ALTD;
OFFQ        ASSIGN:OFF_TIME=1;
            QUEUE,EMERGENCY_ROOM_Q2;
            SEIZE,2:DOCTOR(EMG):
                 NEXT(DEL10);
ALTD        ALTER:DOCTOR(EMG),+1:
                 DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

DEL10       DELAY:PROCESSTIME;
            ASSIGN:DOCT_NO=EMG:
                 EMG=0:NEXT(MERG1);

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; 222222222222222
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
NEMG        BRANCH,1:IF,DAY>5,NWEMG:
                 IF,AMOD(TNOW,1440)>540,OTEMG1:
                 ELSE,RTEMG1;
NWEMG       FINDJ,1,3:NR(DOCTOR(J))==0;
            ASSIGN:EMG1=J;
            BRANCH,1:IF,EMG1==WEEKEND_SURG_NO,
                         EA11:
                 ELSE,EA21;
EA11        ASSIGN:EMG1=WEEKEND_SURG_NO+1;
EA21        ASSIGN:BEGIN_AT_WEEKEND=1;
            BRANCH,2:ALWAYS,WEMG11:
                 ALWAYS,WEMG21;
WEMG11      QUEUE,EMERGENCY_ROOM_Q6;
            SEIZE,2:DOCTOR(EMG1):
                 NEXT(DEL101);
WEMG21      ALTER:DOCTOR(EMG1),+1:
                 DISPOSE;

RTEMG1      QUEUE,EMERGENCY_ROOM_Q4;
            SELECT,RAN:DOCT11:
                 DOCT21:
                 DOCT31;
```

Appendix 12 continued

```
DOCT11     SEIZE,2:DOCTOR(1);
           ASSIGN:DOCT_NO=1:
                EMG1=DOCT_NO:
                NEXT(IFOFF1);
DOCT21     SEIZE,2:DOCTOR(2);
           ASSIGN:DOCT_NO=2:
                EMG1=DOCT_NO:
                NEXT(IFOFF1);
DOCT31     SEIZE,2:DOCTOR(3);
           ASSIGN:DOCT_NO=3:
                EMG1=DOCT_NO:
                NEXT(IFOFF1);
IFOFF1     BRANCH,1:IF,NQ((EMG1+15))==0,IFLUN1:
                ELSE,REMO1;
REMO1      REMOVE:1,15+EMG1,SCRAP:NEXT(IFOFF1);
IFLUN1     BRANCH,1:IF,NQ((EMG1+11))==0,DEL101:
                ELSE,REML1;
REML1      REMOVE:1,EMG+11,SS5:NEXT(IFLUN1);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OTEMG1     BRANCH,1:IF,NR(DOCTOR(1))==1.AND.NR(DOCTOR(2))==1
                .AND.NR(DOCTOR(3))==1;ROFF1:
                ELSE,OFFD1;
ROFF1      ASSIGN:OFF_TIME=1:NEXT(RTEMG1);
OFFD1      ASSIGN:BEGIN_AT_OFF=1;
           FINDJ,1,3:NR(DOCTOR(J))==0;
           ASSIGN:EMG1=J;
           BRANCH,2:ALWAYS,OFFQ1:
                ALWAYS,ALTD1;
OFFQ1      ASSIGN:OFF_TIME=1;
           QUEUE,EMERGENCY_ROOM_Q5;
           SEIZE,2:DOCTOR(EMG1):
                NEXT(DEL101);
ALTD1      ALTER:DOCTOR(EMG1),+1:
                DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

DEL101     DELAY:PROCESSTIME;
           ASSIGN:DOCT_NO=EMG:
                EMG1=0:NEXT(MERG1);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

MERG1      BRANCH,1:IF,AMOD(TNOW,1440)<=240,ST10_1:
                IF,AMOD(TNOW,1440)<=300,ST10_2:
                IF,AMOD(TNOW,1440)<=540,ST10_1:
                IF,AMOD(TNOW,1440)>540,ST10_3;
ST10_1     BRANCH,1:IF,BEGIN_AT_WEEKEND==0.AND.DAY>5,EA4:
                IF,BEGIN_AT_WEEKEND>0.AND.DAY>5,EA4:
```

Appendix 12 continued

```
                IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
                AMOD(TNOW,1440)<480,ST10_2:
                IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
                AMOD(TNOW,1440)>=480,ST10_3:
                ELSE,EA5;
EA4         ALTER:DOCTOR(DOCT_NO),-1;
EA5         RELEASE:DOCTOR(DOCT_NO):NEXT(STABT);
ST10_2      ALTER:DOCTOR(DOCT_NO),-1;
            RELEASE:DOCTOR(DOCT_NO);
            DUPLICATE:1,SCH1:NEXT(STABT);
ST10_3      ALTER:DOCTOR(DOCT_NO),-1;
            RELEASE:DOCTOR(DOCT_NO);
            DUPLICATE:1,SCH2:NEXT(STABT);
SCH1        BRANCH,1:IF,BEGIN_AT_WEEKEND==0.AND.DAY<6,EA3:
                IF,BEGIN_AT_WEEKEND==1.AND.DAY==1,EA3;
EA3         DELAY:60;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            BRANCH,1:IF,BEGIN_AT_WEEKEND==1.AND.DAY==1,MON:
                ELSE,NMON;
MON         DUPLICATE:1,SS5;
NMON        DUPLICATE:1,OPCYCLE;
            DELAY:1;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            ALTER:DOCTOR(DOCT_NO),+1:
                DISPOSE;
SCH2        BRANCH,1:IF,BEGIN_AT_OFF.NE.1.AND.DAY<5.AND.
                BEGIN_AT_WEEKEND==0,
                ST10_4:
                IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
                    AMOD(TNOW,1440)>=480,ST10_4;
ST10_4      DELAY:1439-AMOD(TNOW,1440);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            DUPLICATE:1,OPCYCLE;
            DELAY:2;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            ALTER:DOCTOR(DOCT_NO),+1:
                DISPOSE;
STABT       COUNT:SPE_EMG;
            ASSIGN:NO_OF_SPE_EMG=NO_OF_SPE_EMG+1;
            DELAY:RN(DRG_NO+48)-PROCESSTIME:
                NEXT(PROCT);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

REMG        DELAY:RN(DRG_NO+48);
PROCT       ASSIGN:EXPECT_PAT_PROC_TIME=CO(DRG_NO+9)+
                CO(DRG_NO+12):
                EXPECT_PAT_SURG_TIME=CO(DRG_NO+9);
            FINDJ,1,3:MIN(DOCT_PROC_TIME(J));
            ASSIGN:DOCT_NO=J:
```

Appendix 12 continued

```
                    DOCT_PROC_TIME(J)=DOCT_PROC_TIME(J)
                    +EXPECT_PAT_PROC_TIME;
          RELEASE:EMG_BED;
          ROUTE:0,4;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; QUEUE FOR OPERATING ROOM
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
W_OPER    ASSIGN:W_OPER_TIME=TNOW:
                    2ND_TRY=1:
                    BEG=0;
          BRANCH,1:IF,DOCT_NO==1,BK_OP1:
                    IF,DOCT_NO==2,BK_OP2:
                    IF,DOCT_NO==3,BK_OP3;
BK_OP1    QUEUE,9:DETACH;
BK_OP2    QUEUE,10:DETACH;
BK_OP3    QUEUE,11:DETACH;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OPCYCLE   FINDJ,1,3:MAX(DOCT_SURG_TIME(J));
          ASSIGN:LONGEST=J:
                    LONGEST_TIME=DOCT_SURG_TIME(J):
                    DOCT_SURG_TIME(J)=-1000;
          FINDJ,1,3:MAX(DOCT_SURG_TIME(J));
          ASSIGN:LONGER=J:
                    LONGER_TIME=DOCT_SURG_TIME(J):
                    DOCT_SURG_TIME(J)=-500;
          FINDJ,1,3:MAX(DOCT_SURG_TIME(J));
          ASSIGN:SHORT=J:
                    DOCT_SURG_TIME(LONGEST)=LONGEST_TIME:
                    DOCT_SURG_TIME(LONGER)=LONGER_TIME;
GOHEAD    BRANCH,1:IF,LONGEST==EMG,2NDG:
                    IF,LONGEST==EMG1,2NDG:
                    IF,NQ((8+LONGEST))==0,2NDG:
                    ELSE,1SERG;
1SERG     SEARCH,(8+LONGEST),1,NQ:
                    MAX(PROFIT_MARGIN/PROCESSTIME);
          REMOVE:J,(8+LONGEST),GET_S:
                    DISPOSE;
2NDG      BRANCH,1:IF,LONGER==EMG,3RDG:
                    IF,LONGER==EMG1,3RDG:
                    IF,NQ((8+LONGER))==0,3RDG:
                    ELSE,2SERG;
2SERG     SEARCH,(8+LONGER),1,NQ:
                    MAX(PROFIT_MARGIN/PROCESSTIME);
          REMOVE:J,(8+LONGER),GET_S:
                    DISPOSE;
3RDG
BRANCH,1:IF,NQ((8+SHORT)).NE.0.AND.SHORT.NE.EMG.AND.
                    SHORT.NE.EMG1,3SERG;
3SERG     SEARCH,(8+SHORT),1,NQ:
```

Appendix 12 continued

```
          MAX(PROFIT_MARGIN/PROCESSTIME);
     REMOVE:J,(8+SHORT),GET_S:
          DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; THE TOTAL NO. OF EACH DRG
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          CREATE;
          DELAY:655200;
          ASSIGN:NO_DRG(1)=0:
               NO_DRG(2)=0:
               NO_DRG(3)=0:
               DELAYED_COST=0:
               WAIT_COST=0;
          DELAY:129600;
          COUNT:TOTAL_DRG1,NO_DRG(1);
          COUNT:TOTAL_DRG2,NO_DRG(2);
          COUNT:TOTAL_DRG3,NO_DRG(3);
          COUNT:TOTAL_DELAYED_COST,DELAYED_COST;
          COUNT:TOTAL_WAIT_COST,WAIT_COST:
               DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; SCHEDULE FOR SURGEONS
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          CREATE:1440;
          DELAY:1;
          ASSIGN:TIME_SCHEDULE=1;
          BRANCH,3:ALWAYS,SURG1:
               ALWAYS,SURG2:
               ALWAYS,SURG3;
SURG1     ASSIGN:DOCT_NO=1:NEXT(SSCH);
SURG2     ASSIGN:DOCT_NO=2:NEXT(SSCH);
SURG3     ASSIGN:DOCT_NO=3:NEXT(SSCH);
SSCH      ASSIGN:BEG=1;
          BRANCH,1:IF,DAY<=5,WDAY1:
               IF,DAY==7,WEND1;
WEND1     DELAY:1438:NEXT(REG1);
WDAY1     DELAY:240;
          BRANCH,1:IF,EMG==DOCT_NO,NOLU1:
               IF,EMG1==DOCT_NO,NOLU1:
               ELSE,LU1;
LU1       QUEUE,(DOCT_NO+11);
          SEIZE,3:DOCTOR(DOCT_NO);
          ALTER:DOCTOR(DOCT_NO),-1;
          RELEASE:DOCTOR(DOCT_NO);
          BRANCH,1:IF,AMOD(TNOW,1440)>=480,ODEL1:
               ELSE,LDEL1;
LDEL1     ASSIGN:REST(DOCT_NO)=1;
          DELAY:60;
```

Appendix 12 continued

```
            BRANCH,1:IF,NR(OPERATING_ROOM).NE.1,SS1:
                 ELSE,SS2;
SS1         DUPLICATE:1,OPCYCLE;
SS2         DELAY:1;
            ALTER:DOCTOR(DOCT_NO),+1;
            ASSIGN:REST(DOCT_NO)=0;
SS5         DELAY:542-AMOD(TNOW,1440);
OBRA1       BRANCH,1:IF,EMG.NE.DOCT_NO,OFF1:
                 IF,EMG1.NE.DOCT_NO,OFF1;
OFF1        QUEUE,(15+DOCT_NO);
            SEIZE,3:DOCTOR(DOCT_NO);
            ALTER:DOCTOR(DOCT_NO),-1;
            RELEASE:DOCTOR(DOCT_NO);
ODEL1       ASSIGN:REST(DOCT_NO)=1;
            DELAY:1439-AMOD(TNOW,1440);
            BRANCH,1:IF,DAY.NE.5,REG1;
REG1        DUPLICATE:1,OPCYCLE;
SS3         BRANCH,1:IF,DOCT_NO.NE.EMG.AND.
                 DOCT_NO.NE.EMG1,SS4;
SS4         DELAY:2;
            ASSIGN:REST(DOCT_NO)=0;
            ALTER:DOCTOR(DOCT_NO),+1:
                 DISPOSE;
NOLU1       DELAY:301:NEXT(OBRA1);

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; SCHEDULE FOR WEEKEND SURGEON
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            CREATE:10080;
            ASSIGN:TIME_SCHEDULE=1;
            DELAY:7199;
WFIND       FINDJ,1,3:MAX(DOCT_SURG_TIME(J));
            ASSIGN:WEEKEND_SURG_NO=J;
            BRANCH,1:IF,WEEKEND_SURG_NO==EMG,
                 NWSURG:
                 IF,WEEKEND_SURG_NO==EMG1,
                 NWSURG:
                 ELSE,SAMES;
NWSURG      ASSIGN:WEEKTIME=DOCT_SURG_TIME(WEEKEND_SURG_NO):
                 DOCT_SURG_TIME(WEEKEND_SURG_NO)=-1000;
            FINDJ,1,3:MAX(DOCT_SURG_TIME(J));
            ASSIGN:DOCT_PROC_TIME(WEEKEND_SURG_NO)=WEEKTIME;
            ASSIGN:WEEKEND_SURG_NO=J;
SAMES       BRANCH,1:IF,NQ((8+WEEKEND_SURG_NO))>0,WSEAR3:
                 ELSE,NWS1;
WSEAR3      SEARCH,(8+WEEKEND_SURG_NO),1,NQ:
                 MAX(PROFIT_MARGIN/
                 PROCESSTIME);
            REMOVE:J,(8+WEEKEND_SURG_NO),GET_OP;
NWS1        DELAY:2;
```

Appendix 12 continued

```
          ALTER:DOCTOR(WEEKEND_SURG_NO),+1;
          DELAY:240;
          QUEUE,WEEKEND_LUNCH_SURG_Q;
          SEIZE:DOCTOR(WEEKEND_SURG_NO);
          ALTER:DOCTOR(WEEKEND_SURG_NO),-1;
          RELEASE:DOCTOR(WEEKEND_SURG_NO);
          BRANCH,1:IF,AMOD(TNOW,1440)>=480,WSODEL:
               ELSE,WSLDEL;
WSLDEL    DELAY:60;
          BRANCH,1:IF,NQ((8+WEEKEND_SURG_NO))>0,WSEAR2:
               ELSE,NWS2;
WSEAR2    SEARCH,(8+WEEKEND_SURG_NO),1,NQ:
               MAX(PROFIT_MARGIN/PROCESSTIME);
          REMOVE:J,(8+WEEKEND_SURG_NO),GET_OP;
NWS2      DELAY:1;
          ALTER:DOCTOR(WEEKEND_SURG_NO),+1;
          DELAY:542-AMOD(TNOW,1440);
          QUEUE,WEEKEND_OFF_SURG_Q;
          SEIZE:DOCTOR(WEEKEND_SURG_NO);
          ALTER:DOCTOR(WEEKEND_SURG_NO),-1;
          RELEASE:DOCTOR(WEEKEND_SURG_NO);
WSODEL    DELAY:1439-AMOD(TNOW,1440);
          BRANCH,1:IF,DAY==6,WFIND;
GET_OP    QUEUE,WEEKEND_OPERATING_Q;
          SEIZE,2:OPERATING_ROOM;
          QUEUE,WEEKEND_SURG_Q;
          SEIZE,2:DOCTOR(WEEKEND_SURG_NO):
               NEXT(DEL8);

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;    WEEKDAY CALENDER
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          CREATE:1440;
NEWD      ASSIGN:DAY=DAY+1;
          BRANCH,1:IF,DAY==7,NWEEK;
NWEEK     DELAY:1439;
          ASSIGN:DAY=0:
               DISPOSE;


SCRAP     COUNT:NO_USE:DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;    OVER_NIGHT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          CREATE:1440;
          ASSIGN:CLD=CLD+1;
          BRANCH,1:IF,CLD==1,DD8:
               IF,CLD==7,DD8:
               ELSE,DD7;
```

Appendix 12 continued

```
DD7         ASSIGN:CL=CL+1;
            BRANCH,1:IF,CL<4,DD1:
                 ELSE,DD2;
DD1         BRANCH,1:IF,NQ((CL+11))==0,DD4:
                 ELSE,DD3;
DD3         REMOVE:1,(CL+11),SCRAP:NEXT(DD1);
DD4         BRANCH,1:IF,NQ((CL+15))==0,DD7:
                 ELSE,DD6;
DD6         REMOVE:1,(CL+15),SCRAP:NEXT(DD4);
DD2         ASSIGN:CL=0:DISPOSE;
DD8         BRANCH,1:IF,NQ(15)==0,DD9:
                 ELSE,DD10;
DD10        REMOVE:1,15,SCRAP:NEXT(DD8);
DD9         BRANCH,1:IF,NQ(19)==0,DD13:
                 ELSE,DD11;
DD11        REMOVE:1,19,SCRAP:NEXT(DD9);
DD13        BRANCH,1:IF,CLD==7,DD12;
DD12        ASSIGN:CLD=0:DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;      OFFICE VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            CREATE;
OUTPAT      BRANCH,2:ALWAYS,NEXTO:
                 ALWAYS,NEWO;
NEWO        DELAY:EX(56):NEXT(OUTPAT);
NEXTO       QUEUE,OUT_PAT_Q;
            SELECT,POR:OUT1:
                 OUT2:
                 OUT3;
OUT1        SEIZE,7:DOCTOR(1);
            ASSIGN:DOCT_NO=1:NEXT(OUT4);
OUT2        SEIZE,7:DOCTOR(2);
            ASSIGN:DOCT_NO=2:NEXT(OUT4);
OUT3        SEIZE,7:DOCTOR(3);
            ASSIGN:DOCT_NO=3:NEXT(OUT4);
OUT4        DELAY:CO(52);
            RELEASE:DOCTOR(DOCT_NO):DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;      HOSPITAL RUN
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
HOSP        DELAY:1442-AMOD(TNOW,1440);
            BRANCH,1:IF,IN_HOSP(PAT_NO)==1.AND.DAY<6,SEE;
SEE         QUEUE,HOSP_RUN_Q;
            SEIZE,5:DOCTOR(DOCT_NO);
            DELAY:CO(52+DRG_NO);
            RELEASE:DOCTOR(DOCT_NO):NEXT(HOSP);

END;
```

Appendix 12 continued

```
BEGIN;
            CREATE,3;
            ASSIGN:NUMBER=NUMBER+1:
                 DRG_NO=NUMBER;
            BRANCH,1:IF,DRG_NO<=3,NEW;
NEW         BRANCH,2:ALWAYS,NEXT_S:
                 ALWAYS,W_NEW;
W_NEW       DELAY:EX(DRG_NO,DRG_NO):NEXT(NEW);
NEXT_S      BRANCH,1:WITH,CO(DRG_NO+3),EMG:
                 WITH,(1-CO(DRG_NO+3))*0.7,SUGC:
                 ELSE,SUGNC;
EMG         QUEUE,EMG_BED_Q,0,SCRAP;
            SEIZE:EMG_BED;
            ASSIGN:NO_OF_URGENCE=NO_OF_URGENCE+1:
                 ADMIT_NO=ADMIT_NO+1:
                 PAT_NO=ADMIT_NO:
                 ARRTIME=TNOW:
                 NO_DRG(DRG_NO)=NO_DRG(DRG_NO)+1;
            ROUTE:0,6;
SUGC        ASSIGN:DOCT_NO=DP(DRG_NO+6):
                 EXPECT_PAT_PROC_TIME=CO(DRG_NO+9)
                 +CO(DRG_NO+12):
                 EXPECT_PAT_SURG_TIME=CO(DRG_NO+9):
                 DOCT_PROC_TIME(DOCT_NO)=
                 DOCT_PROC_TIME(DOCT_NO)+
                 EXPECT_PAT_PROC_TIME:
                 NEXT(N_STAT);
SUGNC       ASSIGN:EXPECT_PAT_PROC_TIME=CO(DRG_NO+9)+
                 CO(DRG_NO+12):
                 EXPECT_PAT_SURG_TIME=CO(DRG_NO+9);
            FINDJ,1,3:MIN(DOCT_PROC_TIME(J));
            ASSIGN:DOCT_NO=J:
                 DOCT_PROC_TIME(J)=DOCT_PROC_TIME(J)
                 +EXPECT_PAT_PROC_TIME:
                 NEXT(N_STAT);
N_STAT      ROUTE:0,1;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 1 : FIRST VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT1       STATION,FIRST_VISIT;
            ASSIGN:STAT_NO=1:
                 PROCESSTIME=CO(DRG_NO+15);
GET_OD1     QUEUE,OFFICE_DOCT_Q;
            SEIZE,6:DOCTOR(DOCT_NO);
            DELAY:PROCESSTIME;
            RELEASE:DOCTOR(DOCT_NO);
BAC_1       ROUTE:0,2;
```

Appendix 13 computer program for Concurrent Scheduling model

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 2  DIAGNOSIS INSPECTIONS
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          STATION,INSPECTION;
          BRANCH,1:IF,AMOD(TNOW,1440)>=540,NDAY:
               ELSE,TDAY;
NDAY      DELAY:1441-TNOW;
TDAY      DELAY:300;
          ROUTE:0,3;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 3 FOR POST-DIAGNOSIS VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT6     STATION,FOLLOW_VISIT_STAT;
          ASSIGN:STAT_NO=3:
               PROCESSTIME=CO(DRG_NO+18);
GET_OD6   QUEUE,SAME_DOCT_Q1;
          SEIZE,6:DOCTOR(DOCT_NO);
          DELAY:PROCESSTIME;
          RELEASE:DOCTOR(DOCT_NO);
BAC_6     ROUTE:0,4;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 4 FOR ADMISSION
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT7     STATION,ADMISSION_STAT;
          QUEUE,ADMISSION_Q;
          SEIZE:BED;
          ASSIGN:ADMIT_NO=ADMIT_NO+1:
               PAT_NO=ADMIT_NO:
               IN_HOSP(PAT_NO)=1:
               DOCT_SURG_TIME(DOCT_NO)=
               DOCT_SURG_TIME(DOCT_NO)
               +CO(DRG_NO+9):
               NO_DRG(DRG_NO)=NO_DRG(DRG_NO)+1;

          DUPLICATE:1,HOSP;
          ROUTE:0,5;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 5 FOR SUGERY UNIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT8     STATION,SURGERY_STAT;
          ASSIGN:STAT_NO=5:
               PROCESSTIME=RN(DRG_NO+21):
               LATE_COST=CO(DRG_NO+24):
               WAIT_DUE_LENGTH=CO(DRG_NO+27):
               PROFIT_MARGIN=CO(DRG_NO+30):
               MED_COST=CO(DRG_NO+56):
               ARRTIME=TNOW;
```

Appendix 13 continued

```
          BRANCH,1:IF,DAY>5,W_OPER:
               IF,AMOD(TNOW,1440)<240,GET_S:
               IF,AMOD(TNOW,1440)>302.AND.AMOD(TNOW,1440)
               <540.AND.REST(DOCT_NO)==0,GET_S:
               ELSE,W_OPER;
GET_S     QUEUE,OPERATING_ROOM_Q,0,W_OPER;
          SEIZE:OPERATING_ROOM;
DOCT_Q2   QUEUE,SAME_DOCT_Q2;
          SEIZE:DOCTOR(DOCT_NO):
               NEXT(DEL8);
DEL8      BRANCH,1:IF,DRG_NO==1,ST51:
               IF,DRG_NO==2,ST52:
               IF,DRG_NO==3,ST53;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
ST51      TALLY:WAITING_TIME_FOR_DRG1,TNOW-ARRTIME;
          TALLY:AVERAGE_DRG1_WAITING_TIME,LATE_COST*
               EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
          ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
               EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
               WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
               MED_COST/1440:
               NEXT(ST54);
ST52      TALLY:WAITING_TIME_FOR_DRG2,TNOW-ARRTIME;
          TALLY:AVERAGE_DRG2_WAITING_TIME,LATE_COST*
               EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
          ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
               EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
               WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
               MED_COST/1440:
               NEXT(ST54);
ST53      TALLY:WAITING_TIME_FOR_DRG3,TNOW-ARRTIME;
          TALLY:AVERAGE_DRG3_WAITING_TIME,LATE_COST*
               EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH);
          ASSIGN:DELAYED_COST=DELAYED_COST+LATE_COST*
               EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH):
               WAIT_COST=WAIT_COST+(TNOW-ARRTIME)*
               MED_COST/1440:
               NEXT(ST54);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
ST54      ASSIGN:BEG_SURG_TIME=TNOW;
          ASSIGN:DOCT_PROC_TIME(DOCT_NO)=
               DOCT_PROC_TIME(DOCT_NO)-CO(DRG_NO+9):
               DOCT_SURG_TIME(DOCT_NO)=
               DOCT_SURG_TIME(DOCT_NO)-CO(DRG_NO+9);
          DELAY:PROCESSTIME;                OPERATING ROOM
               BRANCH,1:IF,AMOD(TNOW,1440)>=240.AND.
               NQ((DOCT_NO+11)).NE.0.AND.DAY<=5,RREL:
               IF,AMOD(TNOW,1440)>=240.AND.NQ(15).NE.0
               .AND.DAY>5,RREL:
               IF,AMOD(TNOW,1440)>=540,RREL:
```

Appendix 13 continued

```
                ELSE,DUP8;
RREL        RELEASE:OPERATING_ROOM;
            RELEASE:DOCTOR(DOCT_NO):
                NEXT(RDEL);
DUP8        DUPLICATE:1,FINDB8;
RDEL        BRANCH,1:IF,NOT_ORG==0,ST8_6;
ST8_6       DELAY:CO(DRG_NO+33);                 RECOVERY ROOM
            RELEASE:BED;
            ASSIGN:IN_HOSP(PAT_NO)=0;
            ASSIGN:DOCT_PROC_TIME(DOCT_NO)=
                DOCT_PROC_TIME(DOCT_NO)
                -CO(DRG_NO+12):
                IN_HOSP(PAT_NO)=0;
            BRANCH,1:IF,DRG_NO==1,COUNT1:
                IF,DRG_NO==2,COUNT2:
                IF,DRG_NO==3,COUNT3;
COUNT1      COUNT:SURGERY_OF_DRG1:DISPOSE;
COUNT2      COUNT:SURGERY_OF_DRG2:DISPOSE;
COUNT3      COUNT:SURGERY_OF_DRG3:DISPOSE;
FINDB8      ASSIGN:NOT_ORG=1;
            BRANCH,1:IF,DAY>5,ST8_1:
                ELSE,ST8_2;
ST8_1       BRANCH,1:IF,NQ((8+WEEKEND_SURG_NO))==0,RREL:
                ELSE,S8;
S8          RELEASE:OPERATING_ROOM;
            DUPLICATE:1,REDOC;
            ASSIGN:CURR=WEEKEND_SURG_NO;
            SEARCH,(8+WEEKEND_SURG_NO),1,NQ:
                MAX((PROFIT_MARGIN+LATE_COST*
                EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH))/
                PROCESSTIME);
            REMOVE:J,(8+WEEKEND_SURG_NO),GET_S:
                DISPOSE;
ST8_2       RELEASE:OPERATING_ROOM;
            DUPLICATE:1,REDOC;
            ASSIGN:CURR=DOCT_NO:NEXT(OPCYCLE);
REDOC       DELAY:1;
            RELEASE:DOCTOR(DOCT_NO):DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; STATION 6 FOR EMERGENCE ADMISSION
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
STAT10      STATION,EMERGENCY_ROOM_STAT;
            ASSIGN:STAT_NO=6:
                PROCESSTIME=RN(DRG_NO+36):
                LATE_COST=CO(DRG_NO+39):
                WAIT_DUE_LENGTH=CO(DRG_NO+42):
                PROFIT_MARGIN=CO(DRG_NO+45);
            BRANCH,1:WITH,0.05,SEMG:
                ELSE,REMG;
```

Appendix 13 continued

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
SEMG        BRANCH,1:IF,EMG==0,OEMG:
                  ELSE,NEMG;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; 11111111111111111
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OEMG        BRANCH,1:IF,DAY>5,WEMG:
                  IF,AMOD(TNOW,1440)>540,OTEMG:
                  ELSE,RTEMG;
WEMG        FINDJ,1,3:NR(DOCTOR(J))==0;
            ASSIGN:EMG=J;
            BRANCH,1:IF,EMG==WEEKEND_SURG_NO,
                  EA1:
                  ELSE,EA2;
EA1         ASSIGN:EMG=WEEKEND_SURG_NO+1;
EA2         ASSIGN:BEGIN_AT_WEEKEND=1;
            BRANCH,2:ALWAYS,WEMG1:
                  ALWAYS,WEMG2;
WEMG1       QUEUE,EMERGENCY_ROOM_Q3;
            SEIZE,2:DOCTOR(EMG):
                  NEXT(DEL10);
WEMG2       ALTER:DOCTOR(EMG),+1:
                  DISPOSE;


RTEMG       QUEUE,EMERGENCY_ROOM_Q1;
            SELECT,RAN:DOCT1:
                  DOCT2:
                  DOCT3;
DOCT1       SEIZE,2:DOCTOR(1);
            ASSIGN:DOCT_NO=1:
                  EMG=DOCT_NO:
                  NEXT(IFOFF);
DOCT2       SEIZE,2:DOCTOR(2);
            ASSIGN:DOCT_NO=2:
                  EMG=DOCT_NO:
                  NEXT(IFOFF);
DOCT3       SEIZE,2:DOCTOR(3);
            ASSIGN:DOCT_NO=3:
                  EMG=DOCT_NO:
                  NEXT(IFOFF);
IFOFF       BRANCH,1:IF,NQ((EMG+15))==0,IFLUN:
                  ELSE,REMO;
REMO        REMOVE:1,15+EMG,SCRAP:NEXT(IFOFF);
IFLUN       BRANCH,1:IF,NQ((EMG+11))==0,DEL10:
                  ELSE,REML;
REML        REMOVE:1,EMG+11,SS5:NEXT(IFLUN);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 13 continued

```
OTEMG       BRANCH,1:IF,NR(DOCTOR(1))==1.AND.NR(DOCTOR(2))==1
                 .AND.NR(DOCTOR(3))==1,ROFF:
                 ELSE,OFFD;
ROFF        ASSIGN:OFF_TIME=1:NEXT(RTEMG);
OFFD        ASSIGN:BEGIN_AT_OFF=1;
            FINDJ,1,3:NR(DOCTOR(J))==0;
            ASSIGN:EMG=J;
            BRANCH,2:ALWAYS,OFFQ:
                 ALWAYS,ALTD;
OFFQ        ASSIGN:OFF_TIME=1;
            QUEUE,EMERGENCY_ROOM_Q2;
            SEIZE,2:DOCTOR(EMG):
                 NEXT(DEL10);
ALTD        ALTER:DOCTOR(EMG),+1:
                 DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

DEL10       DELAY:PROCESSTIME;
            ASSIGN:DOCT_NO=EMG:
                 EMG=0:NEXT(MERG1);

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; 222222222222222
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
NEMG        BRANCH,1:IF,DAY>5,NWEMG:
                 IF,AMOD(TNOW,1440)>540,OTEMG1:
                 ELSE,RTEMG1;
NWEMG       FINDJ,1,3:NR(DOCTOR(J))==0;
            ASSIGN:EMG1=J;
            BRANCH,1:IF,EMG1==WEEKEND_SURG_NO,
                      EA11:
                 ELSE,EA21;
EA11        ASSIGN:EMG1=WEEKEND_SURG_NO+1;
EA21        ASSIGN:BEGIN_AT_WEEKEND=1;
            BRANCH,2:ALWAYS,WEMG11:
                 ALWAYS,WEMG21;
WEMG11      QUEUE,EMERGENCY_ROOM_Q6;
            SEIZE,2:DOCTOR(EMG1):
                 NEXT(DEL101);
WEMG21      ALTER:DOCTOR(EMG1),+1:
                 DISPOSE;

RTEMG1      QUEUE,EMERGENCY_ROOM_Q4;
            SELECT,RAN:DOCT11:
                 DOCT21:
                 DOCT31;
DOCT11      SEIZE,2:DOCTOR(1);
            ASSIGN:DOCT_NO=1:
```

Appendix 13 continued

```
                EMG1=DOCT_NO:
                NEXT(IFOFF1);
DOCT21     SEIZE,2:DOCTOR(2);
           ASSIGN:DOCT_NO=2:
                EMG1=DOCT_NO:
                NEXT(IFOFF1);
DOCT31     SEIZE,2:DOCTOR(3);
           ASSIGN:DOCT_NO=3:
                EMG1=DOCT_NO:
                NEXT(IFOFF1);
IFOFF1     BRANCH,1:IF,NQ((EMG1+15))==0,IFLUN1:
                ELSE,REMO1;
REMO1      REMOVE:1,15+EMG1,SCRAP:NEXT(IFOFF1);
IFLUN1     BRANCH,1:IF,NQ((EMG1+11))==0,DEL101:
                ELSE,REML1;
REML1      REMOVE:1,EMG+11,SS5:NEXT(IFLUN1);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OTEMG1     BRANCH,1:IF,NR(DOCTOR(1))==1.AND.NR(DOCTOR(2))==1
                .AND.NR(DOCTOR(3))==1,ROFF1:
                ELSE,OFFD1;
ROFF1      ASSIGN:OFF_TIME=1:NEXT(RTEMG1);
OFFD1      ASSIGN:BEGIN_AT_OFF=1;
           FINDJ,1,3:NR(DOCTOR(J))==0;
           ASSIGN:EMG1=J;
           BRANCH,2:ALWAYS,OFFQ1:
                ALWAYS,ALTD1;
OFFQ1      ASSIGN:OFF_TIME=1;
           QUEUE,EMERGENCY_ROOM_Q5;
           SEIZE,2:DOCTOR(EMG1):
                NEXT(DEL101);
ALTD1      ALTER:DOCTOR(EMG1),+1:
                DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

DEL101     DELAY:PROCESSTIME;
           ASSIGN:DOCT_NO=EMG:
                EMG1=0:NEXT(MERG1);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

MERG1      BRANCH,1:IF,AMOD(TNOW,1440)<=240,ST10_1:
                IF,AMOD(TNOW,1440)<=300,ST10_2:
                IF,AMOD(TNOW,1440)<=540,ST10_1:
                IF,AMOD(TNOW,1440)>540,ST10_3;
ST10_1     BRANCH,1:IF,BEGIN_AT_WEEKEND==0.AND.DAY>5,EA4:
                IF,BEGIN_AT_WEEKEND>0.AND.DAY>5,EA4:
                IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
                AMOD(TNOW,1440)<480,ST10_2:
```

Appendix 13 continued

```
                IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
                AMOD(TNOW,1440)>=480,ST10_3:
                ELSE,EA5;
EA4        ALTER:DOCTOR(DOCT_NO),-1;
EA5        RELEASE:DOCTOR(DOCT_NO):NEXT(STABT);
ST10_2     ALTER:DOCTOR(DOCT_NO),-1;
           RELEASE:DOCTOR(DOCT_NO);
           DUPLICATE:1,SCH1:NEXT(STABT);
ST10_3     ALTER:DOCTOR(DOCT_NO),-1;
           RELEASE:DOCTOR(DOCT_NO);
           DUPLICATE:1,SCH2:NEXT(STABT);
SCH1       BRANCH,1:IF,BEGIN_AT_WEEKEND==0.AND.DAY<6,EA3:
                IF,BEGIN_AT_WEEKEND==1.AND.DAY==1,EA3;
EA3        DELAY:60;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
           BRANCH,1:IF,BEGIN_AT_WEEKEND==1.AND.DAY==1,MON:
                ELSE,NMON;
MON        DUPLICATE:1,SS5;
NMON       DUPLICATE:1,OPCYCLE;
           DELAY:1;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
           ALTER:DOCTOR(DOCT_NO),+1:
                DISPOSE;
SCH2       BRANCH,1:IF,BEGIN_AT_OFF.NE.1.AND.DAY<5.AND.
                BEGIN_AT_WEEKEND==0,
                ST10_4:
                IF,BEGIN_AT_WEEKEND>0.AND.DAY==1.AND.
                     AMOD(TNOW,1440)>=480,ST10_4;
ST10_4     DELAY:1439-AMOD(TNOW,1440);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
           DUPLICATE:1,OPCYCLE;
           DELAY:2;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
           ALTER:DOCTOR(DOCT_NO),+1:
                DISPOSE;
STABT      COUNT:SPE_EMG;
           ASSIGN:NO_OF_SPE_EMG=NO_OF_SPE_EMG+1;
           DELAY:RN(DRG_NO+48)-PROCESSTIME:
                NEXT(PROCT);
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

REMG       DELAY:RN(DRG_NO+48);
PROCT      ASSIGN:EXPECT_PAT_PROC_TIME=CO(DRG_NO+9)+
                CO(DRG_NO+12):
                EXPECT_PAT_SURG_TIME=CO(DRG_NO+9);
           FINDJ,1,3:MIN(DOCT_PROC_TIME(J));
           ASSIGN:DOCT_NO=J:
                DOCT_PROC_TIME(J)=DOCT_PROC_TIME(J)
                +EXPECT_PAT_PROC_TIME;
```

Appendix 13 continued

```
          RELEASE:EMG_BED;
          ROUTE:0,4;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; QUEUE FOR OPERATING ROOM
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
W_OPER    ASSIGN:W_OPER_TIME=TNOW:
               2ND_TRY=1:
               BEG=0;
          BRANCH,1:IF,DOCT_NO==1,BK_OP1:
               IF,DOCT_NO==2,BK_OP2:
               IF,DOCT_NO==3,BK_OP3;
BK_OP1    QUEUE,9:DETACH;
BK_OP2    QUEUE,10:DETACH;
BK_OP3    QUEUE,11:DETACH;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
OPCYCLE   FINDJ,1,3:MAX(DOCT_SURG_TIME(J));
          ASSIGN:LONGEST=J:
               LONGEST_TIME=DOCT_SURG_TIME(J):
               DOCT_SURG_TIME(J)=-1000;
          FINDJ,1,3:MAX(DOCT_SURG_TIME(J));
          ASSIGN:LONGER=J:
               LONGER_TIME=DOCT_SURG_TIME(J):
               DOCT_SURG_TIME(J)=-500;
          FINDJ,1,3:MAX(DOCT_SURG_TIME(J));
          ASSIGN:SHORT=J:
               DOCT_SURG_TIME(LONGEST)=LONGEST_TIME:
               DOCT_SURG_TIME(LONGER)=LONGER_TIME;
GOHEAD    BRANCH,1:IF,LONGEST==EMG,2NDG:
               IF,LONGEST==EMG1,2NDG:
               IF,NQ((8+LONGEST))==0,2NDG:
               ELSE,1SERG;
1SERG     SEARCH,(8+LONGEST),1,NQ:
               MAX((PROFIT_MARGIN+LATE_COST*
               EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH))/
               PROCESSTIME);
          REMOVE:J,(8+LONGEST),GET_S:
               DISPOSE;
2NDG      BRANCH,1:IF,LONGER==EMG,3RDG:
               IF,LONGER==EMG1,3RDG:
               IF,NQ((8+LONGER))==0,3RDG:
               ELSE,2SERG;
2SERG     SEARCH,(8+LONGER),1,NQ:
               MAX((PROFIT_MARGIN+LATE_COST*
               EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH))/
               PROCESSTIME);
          REMOVE:J,(8+LONGER),GET_S:
               DISPOSE;
3RDG      BRANCH,1:IF,NQ((8+SHORT)).NE.0.AND.SHORT.NE.EMG
               .AND.SHORT.NE.EMG1,3SERG;
```

Appendix 13 continued

```
3SERG       SEARCH,(8+SHORT),1,NQ:
                 MAX((PROFIT_MARGIN+LATE_COST*
                 EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH))/
                 PROCESSTIME);
            REMOVE:J,(8+SHORT),GET_S:
                 DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; THE TOTAL NO. OF EACH DRG
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            CREATE;
            DELAY:655200;
            ASSIGN:NO_DRG(1)=0:
                 NO_DRG(2)=0:
                 NO_DRG(3)=0:
                 DELAYED_COST=0:
                 WAIT_COST=0;
            DELAY:129600;
            COUNT:TOTAL_DRG1,NO_DRG(1);
            COUNT:TOTAL_DRG2,NO_DRG(2);
            COUNT:TOTAL_DRG3,NO_DRG(3);
            COUNT:TOTAL_DELAYED_COST,DELAYED_COST;
            COUNT:TOTAL_WAIT_COST,WAIT_COST:
                 DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; SCHEDULE FOR SURGEONS
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            CREATE:1440;
            DELAY:1;
            ASSIGN:TIME_SCHEDULE=1;
            BRANCH,3:ALWAYS,SURG1:
                 ALWAYS,SURG2:
                 ALWAYS,SURG3;
SURG1       ASSIGN:DOCT_NO=1:NEXT(SSCH);
SURG2       ASSIGN:DOCT_NO=2:NEXT(SSCH);
SURG3       ASSIGN:DOCT_NO=3:NEXT(SSCH);
SSCH        ASSIGN:BEG=1;
            BRANCH,1:IF,DAY<=5,WDAY1:
                 IF,DAY==7,WEND1;
WEND1       DELAY:1438:NEXT(REG1);
WDAY1       DELAY:240;
            BRANCH,1:IF,EMG==DOCT_NO,NOLU1:
                 IF,EMG1==DOCT_NO,NOLU1:
                 ELSE,LU1;
LU1         QUEUE,(DOCT_NO+11);
            SEIZE,3:DOCTOR(DOCT_NO);
            ALTER:DOCTOR(DOCT_NO),-1;
            RELEASE:DOCTOR(DOCT_NO);
```

Appendix 13 continued

```
          BRANCH,1:IF,AMOD(TNOW,1440)>=480,ODEL1:
               ELSE,LDEL1;
LDEL1     ASSIGN:REST(DOCT_NO)=1;
          DELAY:60;
          BRANCH,1:IF,NR(OPERATING_ROOM).NE.1,SS1:
               ELSE,SS2;
SS1       DUPLICATE:1,OPCYCLE;
SS2       DELAY:1;
          ALTER:DOCTOR(DOCT_NO),+1;
          ASSIGN:REST(DOCT_NO)=0;
SS5       DELAY:542-AMOD(TNOW,1440);
OBRA1     BRANCH,1:IF,EMG.NE.DOCT_NO,OFF1:
               IF,EMG1.NE.DOCT_NO,OFF1;
OFF1      QUEUE,(15+DOCT_NO);
          SEIZE,3:DOCTOR(DOCT_NO);
          ALTER:DOCTOR(DOCT_NO),-1;
          RELEASE:DOCTOR(DOCT_NO);
ODEL1     ASSIGN:REST(DOCT_NO)=1;
          DELAY:1439-AMOD(TNOW,1440);
          BRANCH,1:IF,DAY.NE.5,REG1;
REG1      DUPLICATE:1,OPCYCLE;
SS3       BRANCH,1:IF,DOCT_NO.NE.EMG.AND.
               DOCT_NO.NE.EMG1,SS4;
SS4       DELAY:2;
          ASSIGN:REST(DOCT_NO)=0;
          ALTER:DOCTOR(DOCT_NO),+1:
               DISPOSE;
NOLU1     DELAY:301:NEXT(OBRA1);

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;; SCHEDULE FOR WEEKEND SURGEON
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
          CREATE:10080;
          ASSIGN:TIME_SCHEDULE=1;
          DELAY:7199;
WFIND     FINDJ,1,3:MAX(DOCT_SURG_TIME(J));
          ASSIGN:WEEKEND_SURG_NO=J;
          BRANCH,1:IF,WEEKEND_SURG_NO==EMG,
               NWSURG:
               IF,WEEKEND_SURG_NO==EMG1,
               NWSURG:
               ELSE,SAMES;
NWSURG    ASSIGN:WEEKTIME=DOCT_SURG_TIME(WEEKEND_SURG_NO):
               DOCT_SURG_TIME(WEEKEND_SURG_NO)=-1000;
          FINDJ,1,3:MAX(DOCT_SURG_TIME(J));
          ASSIGN:DOCT_PROC_TIME(WEEKEND_SURG_NO)=WEEKTIME;
          ASSIGN:WEEKEND_SURG_NO=J;
SAMES     BRANCH,1:IF,NQ((8+WEEKEND_SURG_NO))>0,WSEAR3:
               ELSE,NWS1;
```

Appendix 13 continued

```
WSEAR3      SEARCH,(8+WEEKEND_SURG_NO),1,NQ:
                 MAX(PROFIT_MARGIN/
                 PROCESSTIME);
            REMOVE:J,(8+WEEKEND_SURG_NO),GET_OP;
NWS1        DELAY:2;
            ALTER:DOCTOR(WEEKEND_SURG_NO),+1;
            DELAY:240;
            QUEUE,WEEKEND_LUNCH_SURG_Q;
            SEIZE:DOCTOR(WEEKEND_SURG_NO);
            ALTER:DOCTOR(WEEKEND_SURG_NO),-1;
            RELEASE:DOCTOR(WEEKEND_SURG_NO);
            BRANCH,1:IF,AMOD(TNOW,1440)>=480,WSODEL:
                 ELSE,WSLDEL;
WSLDEL      DELAY:60;
            BRANCH,1:IF,NQ((8+WEEKEND_SURG_NO))>0,WSEAR2:
                 ELSE,NWS2;
WSEAR2      SEARCH,(8+WEEKEND_SURG_NO),1,NQ:
                 MAX((PROFIT_MARGIN+LATE_COST*
                 EP((TNOW-ARRTIME)/WAIT_DUE_LENGTH))/
                 PROCESSTIME);
            REMOVE:J,(8+WEEKEND_SURG_NO),GET_OP;
NWS2        DELAY:1;
            ALTER:DOCTOR(WEEKEND_SURG_NO),+1;
            DELAY:542-AMOD(TNOW,1440);
            QUEUE,WEEKEND_OFF_SURG_Q;
            SEIZE:DOCTOR(WEEKEND_SURG_NO);
            ALTER:DOCTOR(WEEKEND_SURG_NO),-1;
            RELEASE:DOCTOR(WEEKEND_SURG_NO);
WSODEL      DELAY:1439-AMOD(TNOW,1440);
            BRANCH,1:IF,DAY==6,WFIND;

GET_OP      QUEUE,WEEKEND_OPERATING_Q;
            SEIZE,2:OPERATING_ROOM;
            QUEUE,WEEKEND_SURG_Q;
            SEIZE,2:DOCTOR(WEEKEND_SURG_NO):
                 NEXT(DEL8);

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;      WEEKDAY CALENDER
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
            CREATE:1440;
NEWD        ASSIGN:DAY=DAY+1;
            BRANCH,1:IF,DAY==7,NWEEK;
NWEEK       DELAY:1439;
            ASSIGN:DAY=0:
                 DISPOSE;

SCRAP       COUNT:NO_USE:DISPOSE;
```

Appendix 13 continued

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;     OVER_NIGHT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
           CREATE:1440;
           ASSIGN:CLD=CLD+1;
           BRANCH,1:IF,CLD==1,DD8:
                IF,CLD==7,DD8:
                ELSE,DD7;
DD7        ASSIGN:CL=CL+1;
           BRANCH,1:IF,CL<4,DD1:
                ELSE,DD2;
DD1        BRANCH,1:IF,NQ((CL+11))==0,DD4:
                ELSE,DD3;
DD3        REMOVE:1,(CL+11),SCRAP:NEXT(DD1);
DD4        BRANCH,1:IF,NQ((CL+15))==0,DD7:
                ELSE,DD6;
DD6        REMOVE:1,(CL+15),SCRAP:NEXT(DD4);
DD2        ASSIGN:CL=0:DISPOSE;
DD8        BRANCH,1:IF,NQ(15)==0,DD9:
                ELSE,DD10;
DD10       REMOVE:1,15,SCRAP:NEXT(DD8);
DD9        BRANCH,1:IF,NQ(19)==0,DD13:
                ELSE,DD11;
DD11       REMOVE:1,19,SCRAP:NEXT(DD9);
DD13       BRANCH,1:IF,CLD==7,DD12;
DD12       ASSIGN:CLD=0:DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;     OFFICE VISIT
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
           CREATE;
OUTPAT     BRANCH,2:ALWAYS,NEXTO:
                ALWAYS,NEWO;
NEWO       DELAY:EX(56):NEXT(OUTPAT);
NEXTO      QUEUE,OUT_PAT_Q;
           SELECT,POR:OUT1:
                OUT2:
                OUT3;
OUT1       SEIZE,7:DOCTOR(1);
           ASSIGN:DOCT_NO=1:NEXT(OUT4);
OUT2       SEIZE,7:DOCTOR(2);
           ASSIGN:DOCT_NO=2:NEXT(OUT4);
OUT3       SEIZE,7:DOCTOR(3);
           ASSIGN:DOCT_NO=3:NEXT(OUT4);
OUT4       DELAY:CO(52);
           RELEASE:DOCTOR(DOCT_NO):DISPOSE;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;     HOSPITAL RUN
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 13 continued

```
HOSP      DELAY:1442-AMOD(TNOW,1440);
          BRANCH,1:IF,IN_HOSP(PAT_NO)==1.AND.DAY<6,SEE;
SEE       QUEUE,HOSP_RUN_Q;
          SEIZE,5:DOCTOR(DOCT_NO);
          DELAY:CO(52+DRG_NO);
          RELEASE:DOCTOR(DOCT_NO):NEXT(HOSP);

END;
```

Appendix 13 continued

```
BEGIN;
PROJECT,INPATIENT A1,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
     PAT_NO:
     ARRTIME:
     DOCT_NO:
     EXPECT_PAT_PROC_TIME:
     STAT_NO:
     PROCESSTIME:
     PROFIT_MARGIN:
     LATE_COST:
     WAIT_DUE_LENGTH:
     BEG_SURG_TIME:
     2ND_TRY:
     BEG:
     OFF_TIME:
     W_OPER_TIME:
     TIME_SCHEDULE:
     BEGIN_AT_OFF:
     BEGIN_AT_WEEKEND:
     WEIGHT:
     NOT_ORG:
     MED_COST;
VARIABLES:NUMBER:
     CL:
     CLD:
     DELAYED_COST:
     NO_OF_EMGPAT:
     ADMIT_NO:
     DOCT_PROC_TIME(6):
     CURR:
     NO_PASS:
     LONGEST:
     LONGER:
     SHORT:
     LONGEST_TIME:
     LONGER_TIME:
     NO_DRG(8):
     TOTAL(8):
     DAY:
     WEEKEND_SURG_NO:
     NO_OF_URGENCE:
     WEEKTIME:
     LAST_SURG,1:
     NO_OF_SPE_EMG:
     NO_OF_NC:
     REST(3):
     EMG:
     EMG1:
     CURR_PAT_NO:
```

Appendix 14 data file for model A1

```
     WEEK:
     IN_HOSP(3000):
     WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
     2,SAME_DOCT_Q1:
     3,ADMISSION_Q:
     4,OPERATING_ROOM_Q:
     5,SAME_DOCT_Q2:
     6,EMERGENCY_ROOM_Q1:
     7,EMERGENCY_ROOM_Q2:
     8,EMERGENCY_ROOM_Q3:
     9,WAIT_OPERATING_ROOM_Q1:
     10,WAIT_OPERATING_ROOM_Q2:
     11,WAIT_OPERATING_ROOM_Q3:
     12,LUNCH_DOCT1_Q:
     13,LUNCH_DOCT2_Q:
     14,LUNCH_DOCT3_Q:
     15,WEEKEND_LUNCH_SURG_Q:
     16,OFF_DOCT1_Q:
     17,OFF_DOCT2_Q:
     18,OFF_DOCT3_Q:
     19,WEEKEND_OFF_SURG_Q:
     20,WEEKEND_OPERATING_Q:
     21,WEEKEND_SURG_Q:
     22,WAIT_OPERATING_ROOM_Q:
     23,EMG_BED_Q:
     24,HOSP_RUN_Q:
     25,EMERGENCY_ROOM_Q4:
     26,EMERGENCY_ROOM_Q5:
     27,EMERGENCY_ROOM_Q6:
     28,OUT_PAT_Q;
RESOURCES:BED,36:
     EMG_BED,5:
     DOCTOR(3):
     OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
     2, INSPECTION:
     3, FOLLOW_VISIT_STAT:
     4, ADMISSION_STAT:
     5, SURGERY_STAT:
     6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,2000:
     2,2500:
     3,7000:
     4,0.05:
     5,0.08:
     6,0.12:
     7,0.35,1, 0.7,2, 1,3:
     8,0.35,1, 0.65,2, 1,3:
     9,0.30,1, 0.65,2, 1,3:
```

Appendix 14 continued

```
     10,30:
     11,30:
     12,30:
     13,30:
     14,30:
     15,30:
     16,180,30:
     17,360,60:
     18,600,100:
     19,1:
     20,2:
     21,3:
     22,4320:
     23,7200:
     24,14400:
     25,200,50:
     26,360,80:
     27,420,120:
     28,1:
     29,2:
     30,3:
     31,4320,420:
     32,5760,600:
     33,7200,720:
     34,2:
     35,2.5:
     36,5:
     37,21600:
     38,14400:
     39,7200:
     40,10:
     41,25:
     42,80:
     43,30:
     44,20:
     45,20:
     46,20:
     47,150:
     48,1:
     49,1:
     50,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
```

Appendix 14 continued

```
     SURGERY_OF_DRG3:
     TOTAL_DRG1:
     TOTAL_DRG2:
     TOTAL_DRG3:
     NO_USE:
     TOTAL_DELAYED_COST:
     SPE_EMG:
     TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
     NR(DOCTOR(2)):
     NR(DOCTOR(3)):
     NR(OPERATING_ROOM):
     NR(EMG_BED):
     NR(BED):
     NQ(12):
     NQ(13):
     NQ(14):
     NQ(15):
     NQ(16):
     NQ(17):
     NQ(18):
     NQ(19):
     NQ(20):
     NQ(22):
     NQ(28);
REPLICATE,1,0,784800,,,655200;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 14 continued

```
BEGIN;
PROJECT,INPATIENT A2,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
     PAT_NO:
     ARRTIME:
     DOCT_NO:
     EXPECT_PAT_PROC_TIME:
     STAT_NO:
     PROCESSTIME:
     PROFIT_MARGIN:
     LATE_COST:
     WAIT_DUE_LENGTH:
     BEG_SURG_TIME:
     2ND_TRY:
     BEG:
     OFF_TIME:
     W_OPER_TIME:
     TIME_SCHEDULE:
     BEGIN_AT_OFF:
     BEGIN_AT_WEEKEND:
     WEIGHT:
     NOT_ORG:
     MED_COST;
VARIABLES:NUMBER:
     CL:
     CLD:
     DELAYED_COST:
     NO_OF_EMGPAT:
     ADMIT_NO:
     DOCT_PROC_TIME(6):
     CURR:
     NO_PASS:
     LONGEST:
     LONGER:
     SHORT:
     LONGEST_TIME:
     LONGER_TIME:
     NO_DRG(8):
     TOTAL(8):
     DAY:
     WEEKEND_SURG_NO:
     NO_OF_URGENCE:
     WEEKTIME:
     LAST_SURG,1:
     NO_OF_SPE_EMG:
     NO_OF_NC:
     REST(3):
     EMG:
     EMG1:
     CURR_PAT_NO:
```

Appendix 15 data file for model A2

```
     WEEK:
     IN_HOSP(3000):
     WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
     2,SAME_DOCT_Q1:
     3,ADMISSION_Q:
     4,OPERATING_ROOM_Q:
     5,SAME_DOCT_Q2:
     6,EMERGENCY_ROOM_Q1:
     7,EMERGENCY_ROOM_Q2:
     8,EMERGENCY_ROOM_Q3:
     9,WAIT_OPERATING_ROOM_Q1:
     10,WAIT_OPERATING_ROOM_Q2:
     11,WAIT_OPERATING_ROOM_Q3:
     12,LUNCH_DOCT1_Q:
     13,LUNCH_DOCT2_Q:
     14,LUNCH_DOCT3_Q:
     15,WEEKEND_LUNCH_SURG_Q:
     16,OFF_DOCT1_Q:
     17,OFF_DOCT2_Q:
     18,OFF_DOCT3_Q:
     19,WEEKEND_OFF_SURG_Q:
     20,WEEKEND_OPERATING_Q:
     21,WEEKEND_SURG_Q:
     22,WAIT_OPERATING_ROOM_Q:
     23,EMG_BED_Q:
     24,HOSP_RUN_Q:
     25,EMERGENCY_ROOM_Q4:
     26,EMERGENCY_ROOM_Q5:
     27,EMERGENCY_ROOM_Q6:
     28,OUT_PAT_Q;
RESOURCES:BED,36:
     EMG_BED,5:
     DOCTOR(3):
     OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
     2, INSPECTION:
     3, FOLLOW_VISIT_STAT:
     4, ADMISSION_STAT:
     5, SURGERY_STAT:
     6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,1600:
     2,2300:
     3,6500:
     4,0.05:
     5,0.08:
     6,0.12:
     7,0.35,1, 0.7,2, 1,3:
     8,0.35,1, 0.65,2, 1,3:
     9,0.30,1, 0.65,2, 1,3:
```

Appendix 15 continued

```
     10,30:
     11,30:
     12,30:
     13,30:
     14,30:
     15,30:
     16,180,30:
     17,360,60:
     18,600,100:
     19,1:
     20,2:
     21,3:
     22,4320:
     23,7200:
     24,14400:
     25,200,50:
     26,360,80:
     27,420,120:
     28,1:
     29,2:
     30,3:
     31,4320,420:
     32,5760,600:
     33,7200,720:
     34,2:
     35,2.5:
     36,5:
     37,21600:
     38,14400:
     39,7200:
     40,10:
     41,25:
     42,80:
     43,30:
     44,20:
     45,20:
     46,20:
     47,150:
     48,1:
     49,1:
     50,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
```

Appendix 15 continued

```
      SURGERY_OF_DRG3:
      TOTAL_DRG1:
      TOTAL_DRG2:
      TOTAL_DRG3:
      NO_USE:
      TOTAL_DELAYED_COST:
      SPE_EMG:
      TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
      NR(DOCTOR(2)):
      NR(DOCTOR(3)):
      NR(OPERATING_ROOM):
      NR(EMG_BED):
      NR(BED):
      NQ(4):
      NQ(12):
      NQ(13):
      NQ(14):
      NQ(15):
      NQ(16):
      NQ(17):
      NQ(18):
      NQ(19):
      NQ(20):
      NQ(22):
      NQ(28);
REPLICATE,1,0,784800,,,655200;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 15 continued

```
BEGIN;
PROJECT,INPATIENT A3,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
     PAT_NO:
     ARRTIME:
     DOCT_NO:
     EXPECT_PAT_PROC_TIME:
     STAT_NO:
     PROCESSTIME:
     PROFIT_MARGIN:
     LATE_COST:
     WAIT_DUE_LENGTH:
     BEG_SURG_TIME:
     2ND_TRY:
     BEG:
     OFF_TIME:
     W_OPER_TIME:
     TIME_SCHEDULE:
     BEGIN_AT_OFF:
     BEGIN_AT_WEEKEND:
     WEIGHT:
     NOT_ORG:
     MED_COST;
VARIABLES:NUMBER:
     CL:
     CLD:
     DELAYED_COST:
     NO_OF_EMGPAT:
     ADMIT_NO:
     DOCT_PROC_TIME(6):
     CURR:
     NO_PASS:
     LONGEST:
     LONGER:
     SHORT:
     LONGEST_TIME:
     LONGER_TIME:
     NO_DRG(8):
     TOTAL(8):
     DAY:
     WEEKEND_SURG_NO:
     NO_OF_URGENCE:
     WEEKTIME:
     LAST_SURG,1:
     NO_OF_SPE_EMG:
     NO_OF_NC:
     REST(3):
     EMG:
     EMG1:
     CURR_PAT_NO:
```

Appendix 16 data file for model A3

```
     WEEK:
     IN_HOSP(3000):
     WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
     2,SAME_DOCT_Q1:
     3,ADMISSION_Q:
     4,OPERATING_ROOM_Q:
     5,SAME_DOCT_Q2:
     +6,EMERGENCY_ROOM_Q1:
     7,EMERGENCY_ROOM_Q2:
     8,EMERGENCY_ROOM_Q3:
     9,WAIT_OPERATING_ROOM_Q1:
     10,WAIT_OPERATING_ROOM_Q2:
     11,WAIT_OPERATING_ROOM_Q3:
     12,LUNCH_DOCT1_Q:
     13,LUNCH_DOCT2_Q:
     14,LUNCH_DOCT3_Q:
     15,WEEKEND_LUNCH_SURG_Q:
     16,OFF_DOCT1_Q:
     17,OFF_DOCT2_Q:
     18,OFF_DOCT3_Q:
     19,WEEKEND_OFF_SURG_Q:
     20,WEEKEND_OPERATING_Q:
     21,WEEKEND_SURG_Q:
     22,WAIT_OPERATING_ROOM_Q:
     23,EMG_BED_Q:
     24,HOSP_RUN_Q:
     25,EMERGENCY_ROOM_Q4:
     26,EMERGENCY_ROOM_Q5:
     27,EMERGENCY_ROOM_Q6:
     28,OUT_PAT_Q;
RESOURCES:BED,36:
     EMG_BED,5:
     DOCTOR(3):
     OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
     2, INSPECTION:
     3, FOLLOW_VISIT_STAT:
     4, ADMISSION_STAT:
     5, SURGERY_STAT:
     6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,1600:
     2,2200:
     3,6000:
     4,0.05:
     5,0.08:
     6,0.12:
     7,0.35,1, 0.7,2, 1,3:
     8,0.35,1, 0.65,2, 1,3:
     9,0.30,1, 0.65,2, 1,3:
```

Appendix 16 continued

```
     10,30:
     11,30:
     12,30:
     13,30:
     14,30:
     15,30:
     16,180,30:
     17,360,60:
     18,600,100:
     19,1:
     20,2:
     21,3:
     22,4320:
     23,7200:
     24,14400:
     25,200,50:
     26,360,80:
     27,420,120:
     28,1:
     29,2:
     30,3:
     31,4320,420:
     32,5760,600:
     33,7200,720:
     34,2:
     35,2.5:
     36,5:
     37,21600:
     38,14400:
     39,7200:
     40,10:
     41,25:
     42,80:
     43,30:
     44,20:
     45,20:
     46,20:
     47,150:
     48,1:
     49,1:
     50,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
```

Appendix 16 continued

```
     SURGERY_OF_DRG3:
     TOTAL_DRG1:
     TOTAL_DRG2:
     TOTAL_DRG3:
     NO_USE:
     TOTAL_DELAYED_COST:
     SPE_EMG:
     TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
     NR(DOCTOR(2)):
     NR(DOCTOR(3)):
     NR(OPERATING_ROOM):
     NR(EMG_BED):
     NR(BED):
     NQ(12):
     NQ(13):
     NQ(14):
     NQ(15):
     NQ(16):
     NQ(17):
     NQ(18):
     NQ(19):
     NQ(20):
     NQ(22):
     NQ(28);
REPLICATE,1,0,784800,,,655200;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 16 continued

```
BEGIN;
PROJECT,INPATIENT B1,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
      PAT_NO:
      ARRTIME:
      DOCT_NO:
      EXPECT_PAT_PROC_TIME:
      STAT_NO:
      PROCESSTIME:
      PROFIT_MARGIN:
      LATE_COST:
      WAIT_DUE_LENGTH:
      BEG_SURG_TIME:
      2ND_TRY:
      BEG:
      OFF_TIME:
      W_OPER_TIME:
      TIME_SCHEDULE:
      BEGIN_AT_OFF:
      BEGIN_AT_WEEKEND:
      WEIGHT:
      NOT_ORG:
      MED_COST;
VARIABLES:NUMBER:
      CL:
      CLD:
      DELAYED_COST:
      NO_OF_EMGPAT:
      ADMIT_NO:
      DOCT_PROC_TIME(6):
      CURR:
      NO_PASS:
      LONGEST:
      LONGER:
      SHORT:
      LONGEST_TIME:
      LONGER_TIME:
      NO_DRG(8):
      TOTAL(8):
      DAY:
      WEEKEND_SURG_NO:
      NO_OF_URGENCE:
      WEEKTIME:
      LAST_SURG,1:
      NO_OF_SPE_EMG:
      NO_OF_NC:
      REST(3):
      EMG:
      EMG1:
      CURR_PAT_NO:
```

Appendix 17 data file for model B1

```
     WEEK:
     IN_HOSP(3000):
     WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
     2,SAME_DOCT_Q1:
     3,ADMISSION_Q:
     4,OPERATING_ROOM_Q:
     5,SAME_DOCT_Q2:
     6,EMERGENCY_ROOM_Q1:
     7,EMERGENCY_ROOM_Q2:
     8,EMERGENCY_ROOM_Q3:
     9,WAIT_OPERATING_ROOM_Q1:
     10,WAIT_OPERATING_ROOM_Q2:
     11,WAIT_OPERATING_ROOM_Q3:
     12,LUNCH_DOCT1_Q:
     13,LUNCH_DOCT2_Q:
     14,LUNCH_DOCT3_Q:
     15,WEEKEND_LUNCH_SURG_Q:
     16,OFF_DOCT1_Q:
     17,OFF_DOCT2_Q:
     18,OFF_DOCT3_Q:
     19,WEEKEND_OFF_SURG_Q:
     20,WEEKEND_OPERATING_Q:
     21,WEEKEND_SURG_Q:
     22,WAIT_OPERATING_ROOM_Q:
     23,EMG_BED_Q:
     24,HOSP_RUN_Q:
     25,EMERGENCY_ROOM_Q4:
     26,EMERGENCY_ROOM_Q5:
     27,EMERGENCY_ROOM_Q6:
     28,OUT_PAT_Q;
RESOURCES:BED,36:
     EMG_BED,5:
     DOCTOR(3):
     OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
     2, INSPECTION:
     3, FOLLOW_VISIT_STAT:
     4, ADMISSION_STAT:
     5, SURGERY_STAT:
     6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,2000:
     2,2500:
     3,7000:
     4,0.05:
     5,0.08:
     6,0.12:
     7,0.35,1, 0.7,2, 1,3:
     8,0.35,1, 0.65,2, 1,3:
     9,0.30,1, 0.65,2, 1,3:
```

Appendix 17 continued

```
     10,30:
     11,30:
     12,30:
     13,30:
     14,30:
     15,30:
     16,180,30:
     17,360,60:
     18,600,100:
     19,1:
     20,2:
     21,3:
     22,4320:
     23,7200:
     24,14400:
     25,200,50:
     26,360,80:
     27,420,120:
     28,1:
     29,2:
     30,3:
     31,4320,420:
     32,5760,600:
     33,7200,720:
     34,2:
     35,2.5:
     36,5:
     37,21600:
     38,14400:
     39,7200:
     40,10:
     41,25:
     42,80:
     43,30:
     44,20:
     45,20:
     46,20:
     47,150:
     48,1:
     49,1:
     50,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
```

Appendix 17 continued

```
      SURGERY_OF_DRG3:
      TOTAL_DRG1:
      TOTAL_DRG2:
      TOTAL_DRG3:
      NO_USE:
      TOTAL_DELAYED_COST:
      SPE_EMG:
      TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
      NR(DOCTOR(2)):
      NR(DOCTOR(3)):
      NR(OPERATING_ROOM):
      NR(EMG_BED):
      NR(BED):
      NQ(12):
      NQ(13):
      NQ(14):
      NQ(15):
      NQ(16):
      NQ(17):
      NQ(18):
      NQ(19):
      NQ(20):
      NQ(22):
      NQ(28);
REPLICATE,1,0,784800;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 17 continued

```
BEGIN;
PROJECT,INPATIENT B2,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
      PAT_NO:
      ARRTIME:
      DOCT_NO:
      EXPECT_PAT_PROC_TIME:
      STAT_NO:
      PROCESSTIME:
      PROFIT_MARGIN:
      LATE_COST:
      WAIT_DUE_LENGTH:
      BEG_SURG_TIME:
      2ND_TRY:
      BEG:
      OFF_TIME:
      W_OPER_TIME:
      TIME_SCHEDULE:
      BEGIN_AT_OFF:
      BEGIN_AT_WEEKEND:
      WEIGHT:
      NOT_ORG:
      MED_COST;
VARIABLES:NUMBER:
      CL:
      CLD:
      DELAYED_COST:
      NO_OF_EMGPAT:
      ADMIT_NO:
      DOCT_PROC_TIME(6):
      CURR:
      NO_PASS:
      LONGEST:
      LONGER:
      SHORT:
      LONGEST_TIME:
      LONGER_TIME:
      NO_DRG(8):
      TOTAL(8):
      DAY:
      WEEKEND_SURG_NO:
      NO_OF_URGENCE:
      WEEKTIME:
      LAST_SURG,1:
      NO_OF_SPE_EMG:
      NO_OF_NC:
      REST(3):
      EMG:
      EMG1:
      CURR_PAT_NO:
```

Appendix 18 data file for model B2

```
     WEEK:
     IN_HOSP(3000):
     WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
     2,SAME_DOCT_Q1:
     3,ADMISSION_Q:
     4,OPERATING_ROOM_Q:
     5,SAME_DOCT_Q2:
     6,EMERGENCY_ROOM_Q1:
     7,EMERGENCY_ROOM_Q2:
     8,EMERGENCY_ROOM_Q3:
     9,WAIT_OPERATING_ROOM_Q1:
     10,WAIT_OPERATING_ROOM_Q2:
     11,WAIT_OPERATING_ROOM_Q3:
     12,LUNCH_DOCT1_Q:
     13,LUNCH_DOCT2_Q:
     14,LUNCH_DOCT3_Q:
     15,WEEKEND_LUNCH_SURG_Q:
     16,OFF_DOCT1_Q:
     17,OFF_DOCT2_Q:
     18,OFF_DOCT3_Q:
     19,WEEKEND_OFF_SURG_Q:
     20,WEEKEND_OPERATING_Q:
     21,WEEKEND_SURG_Q:
     22,WAIT_OPERATING_ROOM_Q:
     23,EMG_BED_Q:
     24,HOSP_RUN_Q:
     25,EMERGENCY_ROOM_Q4:
     26,EMERGENCY_ROOM_Q5:
     27,EMERGENCY_ROOM_Q6:
     28,OUT_PAT_Q;
RESOURCES:BED,36:
     EMG_BED,5:
     DOCTOR(3):
     OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
     2, INSPECTION:
     3, FOLLOW_VISIT_STAT:
     4, ADMISSION_STAT:
     5, SURGERY_STAT:
     6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,1600:
     2,2300:
     3,6500:
     4,0.05:
     5,0.08:
     6,0.12:
     7,0.35,1, 0.7,2, 1,3:
     8,0.35,1, 0.65,2, 1,3:
     9,0.30,1, 0.65,2, 1,3:
```

Appendix 18 continued

```
     10,30:
     11,30:
     12,30:
     13,30:
     14,30:
     15,30:
     16,180,30:
     17,360,60:
     18,600,100:
     19,1:
     20,2:
     21,3:
     22,4320:
     23,7200:
     24,14400:
     25,200,50:
     26,360,80:
     27,420,120:
     28,1:
     29,2:
     30,3:
     31,4320,420:
     32,5760,600:
     33,7200,720:
     34,2:
     35,2.5:
     36,5:
     37,21600:
     38,14400:
     39,7200:
     40,10:
     41,25:
     42,80:
     43,30:
     44,20:
     45,20:
     46,20:
     47,150:
     48,1:
     49,1:
     50,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
```

Appendix 18 continued

```
     SURGERY_OF_DRG3:
     TOTAL_DRG1:
     TOTAL_DRG2:
     TOTAL_DRG3:
     NO_USE:
     TOTAL_DELAYED_COST:
     SPE_EMG:
     TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
     NR(DOCTOR(2)):
     NR(DOCTOR(3)):
     NR(OPERATING_ROOM):
     NR(EMG_BED):
     NR(BED):
     NQ(4):
     NQ(12):
     NQ(13):
     NQ(14):
     NQ(15):
     NQ(16):
     NQ(17):
     NQ(18):
     NQ(19):
     NQ(20):
     NQ(22):
     NQ(28);
REPLICATE,1,0,784800,,,655200;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 18 continued

```
BEGIN;
PROJECT,INPATIENT B3,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
     PAT_NO:
     ARRTIME:
     DOCT_NO:
     EXPECT_PAT_PROC_TIME:
     STAT_NO:
     PROCESSTIME:
     PROFIT_MARGIN:
     LATE_COST:
     WAIT_DUE_LENGTH:
     BEG_SURG_TIME:
     2ND_TRY:
     BEG:
     OFF_TIME:
     W_OPER_TIME:
     TIME_SCHEDULE:
     BEGIN_AT_OFF:
     BEGIN_AT_WEEKEND:
     WEIGHT:
     NOT_ORG:
     MED_COST;
VARIABLES:NUMBER:
     CL:
     CLD:
     DELAYED_COST:
     NO_OF_EMGPAT:
     ADMIT_NO:
     DOCT_PROC_TIME(6):
     CURR:
     NO_PASS:
     LONGEST:
     LONGER:
     SHORT:
     LONGEST_TIME:
     LONGER_TIME:
     NO_DRG(8):
     TOTAL(8):
     DAY:
     WEEKEND_SURG_NO:
     NO_OF_URGENCE:
     WEEKTIME:
     LAST_SURG,1:
     NO_OF_SPE_EMG:
     NO_OF_NC:
     REST(3):
     EMG:
     EMG1:
     CURR_PAT_NO:
```

Appendix 19 data file for model B3

```
     WEEK:
     IN_HOSP(3000):
     WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
     2,SAME_DOCT_Q1:
     3,ADMISSION_Q:
     4,OPERATING_ROOM_Q:
     5,SAME_DOCT_Q2:
     6,EMERGENCY_ROOM_Q1:
     7,EMERGENCY_ROOM_Q2:
     8,EMERGENCY_ROOM_Q3:
     9,WAIT_OPERATING_ROOM_Q1:
     10,WAIT_OPERATING_ROOM_Q2:
     11,WAIT_OPERATING_ROOM_Q3:
     12,LUNCH_DOCT1_Q:
     13,LUNCH_DOCT2_Q:
     14,LUNCH_DOCT3_Q:
     15,WEEKEND_LUNCH_SURG_Q:
     16,OFF_DOCT1_Q:
     17,OFF_DOCT2_Q:
     18,OFF_DOCT3_Q:
     19,WEEKEND_OFF_SURG_Q:
     20,WEEKEND_OPERATING_Q:
     21,WEEKEND_SURG_Q:
     22,WAIT_OPERATING_ROOM_Q:
     23,EMG_BED_Q:
     24,HOSP_RUN_Q:
     25,EMERGENCY_ROOM_Q4:
     26,EMERGENCY_ROOM_Q5:
     27,EMERGENCY_ROOM_Q6:
     28,OUT_PAT_Q;
RESOURCES:BED,36:
     EMG_BED,5:
     DOCTOR(3):
     OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
     2, INSPECTION:
     3, FOLLOW_VISIT_STAT:
     4, ADMISSION_STAT:
     5, SURGERY_STAT:
     6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,1600:
     2,2200:
     3,6000:
     4,0.05:
     5,0.08:
     6,0.12:
     7,0.35,1, 0.7,2, 1,3:
     8,0.35,1, 0.65,2, 1,3:
     9,0.30,1, 0.65,2, 1,3:
```

Appendix 19 continued

```
     10,30:
     11,30:
     12,30:
     13,30:
     14,30:
     15,30:
     16,180,30:
     17,360,60:
     18,600,100:
     19,1:
     20,2:
     21,3:
     22,4320:
     23,7200:
     24,14400:
     25,200,50:
     26,360,80:
     27,420,120:
     28,1:
     29,2:
     30,3:
     31,4320,420:
     32,5760,600:
     33,7200,720:
     34,2:
     35,2.5:
     36,5:
     37,21600:
     38,14400:
     39,7200:
     40,10:
     41,25:
     42,80:
     43,30:
     44,20:
     45,20:
     46,20:
     47,150:
     48,1:
     49,1:
     50,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
```

Appendix 19 continued

```
     SURGERY_OF_DRG3:
     TOTAL_DRG1:
     TOTAL_DRG2:
     TOTAL_DRG3:
     NO_USE:
     TOTAL_DELAYED_COST:
     SPE_EMG:
     TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
     NR(DOCTOR(2)):
     NR(DOCTOR(3)):
     NR(OPERATING_ROOM):
     NR(EMG_BED):
     NR(BED):
     NQ(12):
     NQ(13):
     NQ(14):
     NQ(15):
     NQ(16):
     NQ(17):
     NQ(18):
     NQ(19):
     NQ(20):
     NQ(22):
     NQ(28);
REPLICATE,1,0,784800,,,655200;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 19 continued

```
BEGIN;
PROJECT,INPATIENT C1,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
     PAT_NO:
     ARRTIME:
     DOCT_NO:
     EXPECT_PAT_PROC_TIME:
     EXPECT_PAT_SURG_TIME:
     STAT_NO:
     PROCESSTIME:
     PROFIT_MARGIN:
     LATE_COST:
     WAIT_DUE_LENGTH:
     BEG_SURG_TIME:
     2ND_TRY:
     BEG:
     OFF_TIME:
     W_OPER_TIME:
     TIME_SCHEDULE:
     BEGIN_AT_OFF:
     BEGIN_AT_WEEKEND:
     NOT_ORG:
     MED_COST;
VARIABLES:NUMBER:
     CL:
     CLD:
     DELAYED_COST:
     ADMIT_NO:
     DOCT_PROC_TIME(6):
     DOCT_SURG_TIME(6):
     CURR:
     NO_PASS:
     LONGEST:
     LONGER:
     SHORT:
     LONGEST_TIME:
     LONGER_TIME:
     NO_DRG(8):
     TOTAL(8):
     DAY:
     WEEKEND_SURG_NO:
     NO_OF_URGENCE:
     WEEKTIME:
     NO_OF_SPE_EMG:
     REST(3):
     EMG:
     EMG1:
     IN_HOSP(3000):
     WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
```

Appendix 20 data file for model C1

```
     2,SAME_DOCT_Q1:
     3,ADMISSION_Q:
     4,OPERATING_ROOM_Q:
     5,SAME_DOCT_Q2:
     6,EMERGENCY_ROOM_Q1:
     7,EMERGENCY_ROOM_Q2:
     8,EMERGENCY_ROOM_Q3:
     9,WAIT_OPERATING_ROOM_Q1:
     10,WAIT_OPERATING_ROOM_Q2:
     11,WAIT_OPERATING_ROOM_Q3:
     12,LUNCH_DOCT1_Q:
     13,LUNCH_DOCT2_Q:
     14,LUNCH_DOCT3_Q:
     15,WEEKEND_LUNCH_SURG_Q:
     16,OFF_DOCT1_Q:
     17,OFF_DOCT2_Q:
     18,OFF_DOCT3_Q:
     19,WEEKEND_OFF_SURG_Q:
     20,WEEKEND_OPERATING_Q:
     21,WEEKEND_SURG_Q:
     22,EMG_BED_Q:
     23,OUT_PAT_Q:
     24,HOSP_RUN_Q:
     25,EMERGENCY_ROOM_Q4:
     26,EMERGENCY_ROOM_Q5:
     27,EMERGENCY_ROOM_Q6;
RESOURCES:BED,36:
     EMG_BED,5:
     DOCTOR(3):
     OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
     2, INSPECTION:
     3, FOLLOW_VISIT_STAT:
     4, ADMISSION_STAT:
     5, SURGERY_STAT:
     6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,2000:
     2,2500:
     3,7000:
     4,0.05:
     5,0.08:
     6,0.12:
     7,0.35,1, 0.7,2, 1,3:
     8,0.35,1, 0.65,2, 1,3:
     9,0.30,1, 0.65,2, 1,3:
     10,180:
     11,360:
     12,600:
     13,20:
     14,20:
```

Appendix 20 continued

```
     15,20:
     16,30:
     17,30:
     18,30:
     19,30:
     20,30:
     21,30:
     22,180,30:
     23,360,60:
     24,600,100:
     25,2:
     26,2.5:
     27,5:
     28,21600:
     29,14400:
     30,7200:
     31,10:
     32,25:
     33,80:
     34,4320:
     35,7200:
     36,14400:
     37,200,50:
     38,360,80:
     39,420,120:
     40,10000:
     41,25000:
     42,60000:
     43,100:
     44,60:
     45,30:
     46,3000:
     47,7500:
     48,24000:
     49,4320,420:
     50,5760,600:
     51,7200,720:
     52,30:
     53,20:
     54,20:
     55,20:
     56,150:
     57,1:
     58,1:
     59,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
```

Appendix 20 continued

```
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
     SURGERY_OF_DRG3:
     TOTAL_DRG1:
     TOTAL_DRG2:
     TOTAL_DRG3:
     NO_USE:
     TOTAL_DELAYED_COST:
     SPE_EMG:
     TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
     NR(DOCTOR(2)):
     NR(DOCTOR(3)):
     NR(OPERATING_ROOM):
     NR(EMG_BED):
     NR(BED):
     NQ(9):
     NQ(10):
     NQ(11):
     NQ(12):
     NQ(13):
     NQ(14):
     NQ(15):
     NQ(16):
     NQ(17):
     NQ(18):
     NQ(19):
     NQ(20):
     NQ(23);
REPLICATE,1,0,784800;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 20 continued

```
BEGIN;
PROJECT,INPATIENT C2,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
      PAT_NO:
      ARRTIME:
      DOCT_NO:
      EXPECT_PAT_PROC_TIME:
      EXPECT_PAT_SURG_TIME:
      STAT_NO:
      PROCESSTIME:
      PROFIT_MARGIN:
      LATE_COST:
      WAIT_DUE_LENGTH:
      BEG_SURG_TIME:
      2ND_TRY:
      BEG:
      OFF_TIME:
      W_OPER_TIME:
      TIME_SCHEDULE:
      BEGIN_AT_OFF:
      BEGIN_AT_WEEKEND:
      NOT_ORG:
      MED_COST;
VARIABLES:NUMBER:
      CL:
      CLD:
      DELAYED_COST:
      ADMIT_NO:
      DOCT_PROC_TIME(6):
      DOCT_SURG_TIME(6):
      CURR:
      NO_PASS:
      LONGEST:
      LONGER:
      SHORT:
      LONGEST_TIME:
      LONGER_TIME:
      NO_DRG(8):
      TOTAL(8):
      DAY:
      WEEKEND_SURG_NO:
      NO_OF_URGENCE:
      WEEKTIME:
      NO_OF_SPE_EMG:
      REST(3):
      EMG:
      EMG1:
      IN_HOSP(3000):
      WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
```

Appendix 21 data file for model C2

```
     2,SAME_DOCT_Q1:
     3,ADMISSION_Q:
     4,OPERATING_ROOM_Q:
     5,SAME_DOCT_Q2:
     6,EMERGENCY_ROOM_Q1:
     7,EMERGENCY_ROOM_Q2:
     8,EMERGENCY_ROOM_Q3:
     9,WAIT_OPERATING_ROOM_Q1:
     10,WAIT_OPERATING_ROOM_Q2:
     11,WAIT_OPERATING_ROOM_Q3:
     12,LUNCH_DOCT1_Q:
     13,LUNCH_DOCT2_Q:
     14,LUNCH_DOCT3_Q:
     15,WEEKEND_LUNCH_SURG_Q:
     16,OFF_DOCT1_Q:
     17,OFF_DOCT2_Q:
     18,OFF_DOCT3_Q:
     19,WEEKEND_OFF_SURG_Q:
     20,WEEKEND_OPERATING_Q:
     21,WEEKEND_SURG_Q:
     22,EMG_BED_Q:
     23,OUT_PAT_Q:
     24,HOSP_RUN_Q:
     25,EMERGENCY_ROOM_Q4:
     26,EMERGENCY_ROOM_Q5:
     27,EMERGENCY_ROOM_Q6;
RESOURCES:BED,36:
     EMG_BED,5:
     DOCTOR(3):
     OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
     2, INSPECTION:
     3, FOLLOW_VISIT_STAT:
     4, ADMISSION_STAT:
     5, SURGERY_STAT:
     6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,1600:
     2,2300:
     3,6500:
     4,0.05:
     5,0.08:
     6,0.12:
     7,0.35,1, 0.7,2, 1,3:
     8,0.35,1, 0.65,2, 1,3:
     9,0.30,1, 0.65,2, 1,3:
     10,180:
     11,360:
     12,600:
     13,20:
     14,20:
```

Appendix 21 continued

```
     15,20:
     16,30:
     17,30:
     18,30:
     19,30:
     20,30:
     21,30:
     22,180,30:
     23,360,60:
     24,600,100:
     25,2:
     26,2.5:
     27,5:
     28,21600:
     29,14400:
     30,7200:
     31,10:
     32,25:
     33,80:
     34,4320:
     35,7200:
     36,14400:
     37,200,50:
     38,360,80:
     39,420,120:
     40,10000:
     41,25000:
     42,60000:
     43,100:
     44,60:
     45,30:
     46,3000:
     47,7500:
     48,24000:
     49,4320,420:
     50,5760,600:
     51,7200,720:
     52,30:
     53,20:
     54,20:
     55,20:
     56,150:
     57,1:
     58,1:
     59,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
```

Appendix 21 continued

```
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
     SURGERY_OF_DRG3:
     TOTAL_DRG1:
     TOTAL_DRG2:
     TOTAL_DRG3:
     NO_USE:
     TOTAL_DELAYED_COST:
     SPE_EMG:
     TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
     NR(DOCTOR(2)):
     NR(DOCTOR(3)):
     NR(OPERATING_ROOM):
     NR(EMG_BED):
     NR(BED):
     NQ(4):
     NQ(9):
     NQ(10):
     NQ(11):
     NQ(12):
     NQ(13):
     NQ(14):
     NQ(15):
     NQ(16):
     NQ(17):
     NQ(18):
     NQ(19):
     NQ(20):
     NQ(23);
REPLICATE,1,0,784800,,,655200;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 21 continued

```
BEGIN;
PROJECT,INPATIENT C3,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
      PAT_NO:
      ARRTIME:
      DOCT_NO:
      EXPECT_PAT_PROC_TIME:
      EXPECT_PAT_SURG_TIME:
      STAT_NO:
      PROCESSTIME:
      PROFIT_MARGIN:
      LATE_COST:
      WAIT_DUE_LENGTH:
      BEG_SURG_TIME:
      2ND_TRY:
      BEG:
      OFF_TIME:
      W_OPER_TIME:
      TIME_SCHEDULE:
      BEGIN_AT_OFF:
      BEGIN_AT_WEEKEND:
      NOT_ORG:
      MED_COST;
VARIABLES:NUMBER:
      CL:
      CLD:
      DELAYED_COST:
      ADMIT_NO:
      DOCT_PROC_TIME(6):
      DOCT_SURG_TIME(6):
      CURR:
      NO_PASS:
      LONGEST:
      LONGER:
      SHORT:
      LONGEST_TIME:
      LONGER_TIME:
      NO_DRG(8):
      TOTAL(8):
      DAY:
      WEEKEND_SURG_NO:
      NO_OF_URGENCE:
      WEEKTIME:
      NO_OF_SPE_EMG:
      REST(3):
      EMG:
      EMG1:
      IN_HOSP(3000):
      WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
```

Appendix 22 data file for model C3

```
     2,SAME_DOCT_Q1:
     3,ADMISSION_Q:
     4,OPERATING_ROOM_Q:
     5,SAME_DOCT_Q2:
     6,EMERGENCY_ROOM_Q1:
     7,EMERGENCY_ROOM_Q2:
     8,EMERGENCY_ROOM_Q3:
     9,WAIT_OPERATING_ROOM_Q1:
     10,WAIT_OPERATING_ROOM_Q2:
     11,WAIT_OPERATING_ROOM_Q3:
     12,LUNCH_DOCT1_Q:
     13,LUNCH_DOCT2_Q:
     14,LUNCH_DOCT3_Q:
     15,WEEKEND_LUNCH_SURG_Q:
     16,OFF_DOCT1_Q:
     17,OFF_DOCT2_Q:
     18,OFF_DOCT3_Q:
     19,WEEKEND_OFF_SURG_Q:
     20,WEEKEND_OPERATING_Q:
     21,WEEKEND_SURG_Q:
     22,EMG_BED_Q:
     23,OUT_PAT_Q:
     24,HOSP_RUN_Q:
     25,EMERGENCY_ROOM_Q4:
     26,EMERGENCY_ROOM_Q5:
     27,EMERGENCY_ROOM_Q6;
RESOURCES:BED,36:
     EMG_BED,5:
     DOCTOR(3):
     OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
     2, INSPECTION:
     3, FOLLOW_VISIT_STAT:
     4, ADMISSION_STAT:
     5, SURGERY_STAT:
     6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,1550:
     2,2200:
     3,6000:
     4,0.05:
     5,0.08:
     6,0.12:
     7,0.35,1, 0.7,2, 1,3:
     8,0.35,1, 0.65,2, 1,3:
     9,0.30,1, 0.65,2, 1,3:
     10,180:
     11,360:
     12,600:
     13,20:
     14,20:
```

Appendix 22 continued

```
     15,20:
     16,30:
     17,30:
     18,30:
     19,30:
     20,30:
     21,30:
     22,180,30:
     23,360,60:
     24,600,100:
     25,2:
     26,2.5:
     27,5:
     28,21600:
     29,14400:
     30,7200:
     31,10:
     32,25:
     33,80:
     34,4320:
     35,7200:
     36,14400:
     37,200,50:
     38,360,80:
     39,420,120:
     40,10000:
     41,25000:
     42,60000:
     43,100:
     44,60:
     45,30:
     46,3000:
     47,7500:
     48,24000:
     49,4320,420:
     50,5760,600:
     51,7200,720:
     52,30:
     53,20:
     54,20:
     55,20:
     56,150:
     57,1:
     58,1:
     59,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
```

Appendix 22 continued

```
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
     SURGERY_OF_DRG3:
     TOTAL_DRG1:
     TOTAL_DRG2:
     TOTAL_DRG3:
     NO_USE:
     TOTAL_DELAYED_COST:
     SPE_EMG:
     TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
     NR(DOCTOR(2)):
     NR(DOCTOR(3)):
     NR(OPERATING_ROOM):
     NR(EMG_BED):
     NR(BED):
     NQ(9):
     NQ(10):
     NQ(11):
     NQ(12):
     NQ(13):
     NQ(14):
     NQ(15):
     NQ(16):
     NQ(17):
     NQ(18):
     NQ(19):
     NQ(20):
     NQ(23);
REPLICATE,1,0,784800,,,655200;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 22 continued

```
BEGIN;
PROJECT,INPATIENT D1,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
     PAT_NO:
     ARRTIME:
     DOCT_NO:
     EXPECT_PAT_PROC_TIME:
     EXPECT_PAT_SURG_TIME:
     STAT_NO:
     PROCESSTIME:
     PROFIT_MARGIN:
     LATE_COST:
     WAIT_DUE_LENGTH:
     BEG_SURG_TIME:
     2ND_TRY:
     BEG:
     OFF_TIME:
     W_OPER_TIME:
     TIME_SCHEDULE:
     BEGIN_AT_OFF:
     BEGIN_AT_WEEKEND:
     NOT_ORG:
     MED_COST;
VARIABLES:NUMBER:
     CL:
     CLD:
     DELAYED_COST:
     ADMIT_NO:
     DOCT_PROC_TIME(6):
     DOCT_SURG_TIME(6):
     CURR:
     NO_PASS:
     LONGEST:
     LONGER:
     SHORT:
     LONGEST_TIME:
     LONGER_TIME:
     NO_DRG(8):
     TOTAL(8):
     DAY:
     WEEKEND_SURG_NO:
     NO_OF_URGENCE:
     WEEKTIME:
     NO_OF_SPE_EMG:
     REST(3):
     EMG:
     EMG1:
     IN_HOSP(3000):
     WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
```

Appendix 23 data file for model D1

```
     2,SAME_DOCT_Q1:
     3,ADMISSION_Q:
     4,OPERATING_ROOM_Q:
     5,SAME_DOCT_Q2:
     6,EMERGENCY_ROOM_Q1:
     7,EMERGENCY_ROOM_Q2:
     8,EMERGENCY_ROOM_Q3:
     9,WAIT_OPERATING_ROOM_Q1:
     10,WAIT_OPERATING_ROOM_Q2:
     11,WAIT_OPERATING_ROOM_Q3:
     12,LUNCH_DOCT1_Q:
     13,LUNCH_DOCT2_Q:
     14,LUNCH_DOCT3_Q:
     15,WEEKEND_LUNCH_SURG_Q:
     16,OFF_DOCT1_Q:
     17,OFF_DOCT2_Q:
     18,OFF_DOCT3_Q:
     19,WEEKEND_OFF_SURG_Q:
     20,WEEKEND_OPERATING_Q:
     21,WEEKEND_SURG_Q:
     22,EMG_BED_Q:
     23,OUT_PAT_Q:
     24,HOSP_RUN_Q:
     25,EMERGENCY_ROOM_Q4:
     26,EMERGENCY_ROOM_Q5:
     27,EMERGENCY_ROOM_Q6;
RESOURCES:BED,36:
     EMG_BED,5:
     DOCTOR(3):
     OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
     2, INSPECTION:
     3, FOLLOW_VISIT_STAT:
     4, ADMISSION_STAT:
     5, SURGERY_STAT:
     6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,2000:
     2,2500:
     3,7000:
     4,0.05:
     5,0.08:
     6,0.12:
     7,0.35,1, 0.7,2, 1,3:
     8,0.35,1, 0.65,2, 1,3:
     9,0.30,1, 0.65,2, 1,3:
     10,180:
     11,360:
     12,600:
     13,20:
     14,20:
```

Appendix 23 continued

```
     15,20:
     16,30:
     17,30:
     18,30:
     19,30:
     20,30:
     21,30:
     22,180,30:
     23,360,60:
     24,600,100:
     25,2:
     26,2.5:
     27,5:
     28,21600:
     29,14400:
     30,7200:
     31,10:
     32,25:
     33,80:
     34,4320:
     35,7200:
     36,14400:
     37,200,50:
     38,360,80:
     39,420,120:
     40,10000:
     41,25000:
     42,60000:
     43,100:
     44,60:
     45,30:
     46,3000:
     47,7500:
     48,24000:
     49,4320,420:
     50,5760,600:
     51,7200,720:
     52,30:
     53,20:
     54,20:
     55,20:
     56,150:
     57,1:
     58,1:
     59,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
```

Appendix 23 continued

```
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
     SURGERY_OF_DRG3:
     TOTAL_DRG1:
     TOTAL_DRG2:
     TOTAL_DRG3:
     NO_USE:
     TOTAL_DELAYED_COST:
     SPE_EMG:
     TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
     NR(DOCTOR(2)):
     NR(DOCTOR(3)):
     NR(OPERATING_ROOM):
     NR(EMG_BED):
     NR(BED):
     NQ(9):
     NQ(10):
     NQ(11):
     NQ(12):
     NQ(13):
     NQ(14):
     NQ(15):
     NQ(16):
     NQ(17):
     NQ(18):
     NQ(19):
     NQ(20):
     NQ(23);
REPLICATE,1,0,784800;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 23 continued

```
BEGIN;
PROJECT,INPATIENT D2,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
     PAT_NO:
     ARRTIME:
     DOCT_NO:
     EXPECT_PAT_PROC_TIME:
     EXPECT_PAT_SURG_TIME:
     STAT_NO:
     PROCESSTIME:
     PROFIT_MARGIN:
     LATE_COST:
     WAIT_DUE_LENGTH:
     BEG_SURG_TIME:
     2ND_TRY:
     BEG:
     OFF_TIME:
     W_OPER_TIME:
     TIME_SCHEDULE:
     BEGIN_AT_OFF:
     BEGIN_AT_WEEKEND:
     NOT_ORG:
     MED_COST;
VARIABLES:NUMBER:
     CL:
     CLD:
     DELAYED_COST:
     ADMIT_NO:
     DOCT_PROC_TIME(6):
     DOCT_SURG_TIME(6):
     CURR:
     NO_PASS:
     LONGEST:
     LONGER:
     SHORT:
     LONGEST_TIME:
     LONGER_TIME:
     NO_DRG(8):
     TOTAL(8):
     DAY:
     WEEKEND_SURG_NO:
     NO_OF_URGENCE:
     WEEKTIME:
     NO_OF_SPE_EMG:
     REST(3):
     EMG:
     EMG1:
     IN_HOSP(3000):
     WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
```

Appendix 24 data file for model D2

```
     2,SAME_DOCT_Q1:
     3,ADMISSION_Q:
     4,OPERATING_ROOM_Q:
     5,SAME_DOCT_Q2:
     6,EMERGENCY_ROOM_Q1:
     7,EMERGENCY_ROOM_Q2:
     8,EMERGENCY_ROOM_Q3:
     9,WAIT_OPERATING_ROOM_Q1:
     10,WAIT_OPERATING_ROOM_Q2:
     11,WAIT_OPERATING_ROOM_Q3:
     12,LUNCH_DOCT1_Q:
     13,LUNCH_DOCT2_Q:
     14,LUNCH_DOCT3_Q:
     15,WEEKEND_LUNCH_SURG_Q:
     16,OFF_DOCT1_Q:
     17,OFF_DOCT2_Q:
     18,OFF_DOCT3_Q:
     19,WEEKEND_OFF_SURG_Q:
     20,WEEKEND_OPERATING_Q:
     21,WEEKEND_SURG_Q:
     22,EMG_BED_Q:
     23,OUT_PAT_Q:
     24,HOSP_RUN_Q:
     25,EMERGENCY_ROOM_Q4:
     26,EMERGENCY_ROOM_Q5:
     27,EMERGENCY_ROOM_Q6;
RESOURCES:BED,36:
     EMG_BED,5:
     DOCTOR(3):
     OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
     2, INSPECTION:
     3, FOLLOW_VISIT_STAT:
     4, ADMISSION_STAT:
     5, SURGERY_STAT:
     6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,1600:
     2,2300:
     3,6500:
     4,0.05:
     5,0.08:
     6,0.12:
     7,0.35,1, 0.7,2, 1,3:
     8,0.35,1, 0.65,2, 1,3:
     9,0.30,1, 0.65,2, 1,3:
     10,180:
     11,360:
     12,600:
     13,20:
     14,20:
```

Appendix 24 continued

```
     15,20:
     16,30:
     17,30:
     18,30:
     19,30:
     20,30:
     21,30:
     22,180,30:
     23,360,60:
     24,600,100:
     25,2:
     26,2.5:
     27,5:
     28,21600:
     29,14400:
     30,7200:
     31,10:
     32,25:
     33,80:
     34,4320:
     35,7200:
     36,14400:
     37,200,50:
     38,360,80:
     39,420,120:
     40,10000:
     41,25000:
     42,60000:
     43,100:
     44,60:
     45,30:
     46,3000:
     47,7500:
     48,24000:
     49,4320,420:
     50,5760,600:
     51,7200,720:
     52,30:
     53,20:
     54,20:
     55,20:
     56,150:
     57,1:
     58,1:
     59,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
```

Appendix 24 continued

```
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
     SURGERY_OF_DRG3:
     TOTAL_DRG1:
     TOTAL_DRG2:
     TOTAL_DRG3:
     NO_USE:
     TOTAL_DELAYED_COST:
     SPE_EMG:
     TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
     NR(DOCTOR(2)):
     NR(DOCTOR(3)):
     NR(OPERATING_ROOM):
     NR(EMG_BED):
     NR(BED):
     NQ(4):
     NQ(9):
     NQ(10):
     NQ(11):
     NQ(12):
     NQ(13):
     NQ(14):
     NQ(15):
     NQ(16):
     NQ(17):
     NQ(18):
     NQ(19):
     NQ(20):
     NQ(23);
REPLICATE,1,0,784800,,,655200;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 24 continued

```
BEGIN;
PROJECT,INPATIENT D3,WEN_YEN CHEN;
ATTRIBUTES:DRG_NO:
     PAT_NO:
     ARRTIME:
     DOCT_NO:
     EXPECT_PAT_PROC_TIME:
     EXPECT_PAT_SURG_TIME:
     STAT_NO:
     PROCESSTIME:
     PROFIT_MARGIN:
     LATE_COST:
     WAIT_DUE_LENGTH:
     BEG_SURG_TIME:
     2ND_TRY:
     BEG:
     OFF_TIME:
     W_OPER_TIME:
     TIME_SCHEDULE:
     BEGIN_AT_OFF:
     BEGIN_AT_WEEKEND:
     NOT_ORG:
     MED_COST;
VARIABLES:NUMBER:
     CL:
     CLD:
     DELAYED_COST:
     ADMIT_NO:
     DOCT_PROC_TIME(6):
     DOCT_SURG_TIME(6):
     CURR:
     NO_PASS:
     LONGEST:
     LONGER:
     SHORT:
     LONGEST_TIME:
     LONGER_TIME:
     NO_DRG(8):
     TOTAL(8):
     DAY:
     WEEKEND_SURG_NO:
     NO_OF_URGENCE:
     WEEKTIME:
     NO_OF_SPE_EMG:
     REST(3):
     EMG:
     EMG1:
     IN_HOSP(3000):
     WAIT_COST;
QUEUES:1,OFFICE_DOCT_Q:
```

Appendix 25 data file for model D3

```
      2,SAME_DOCT_Q1:
      3,ADMISSION_Q:
      4,OPERATING_ROOM_Q:
      5,SAME_DOCT_Q2:
      6,EMERGENCY_ROOM_Q1:
      7,EMERGENCY_ROOM_Q2:
      8,EMERGENCY_ROOM_Q3:
      9,WAIT_OPERATING_ROOM_Q1:
      10,WAIT_OPERATING_ROOM_Q2:
      11,WAIT_OPERATING_ROOM_Q3:
      12,LUNCH_DOCT1_Q:
      13,LUNCH_DOCT2_Q:
      14,LUNCH_DOCT3_Q:
      15,WEEKEND_LUNCH_SURG_Q:
      16,OFF_DOCT1_Q:
      17,OFF_DOCT2_Q:
      18,OFF_DOCT3_Q:
      19,WEEKEND_OFF_SURG_Q:
      20,WEEKEND_OPERATING_Q:
      21,WEEKEND_SURG_Q:
      22,EMG_BED_Q:
      23,OUT_PAT_Q:
      24,HOSP_RUN_Q:
      25,EMERGENCY_ROOM_Q4:
      26,EMERGENCY_ROOM_Q5:
      27,EMERGENCY_ROOM_Q6;
RESOURCES:BED,36:
      EMG_BED,5:
      DOCTOR(3):
      OPERATING_ROOM;
STATIONS:1, FIRST_VISIT:
      2, INSPECTION:
      3, FOLLOW_VISIT_STAT:
      4, ADMISSION_STAT:
      5, SURGERY_STAT:
      6, EMERGENCY_ROOM_STAT;
PARAMETERS:1,1600:
      2,2200:
      3,6000:
      4,0.05:
      5,0.08:
      6,0.12:
      7,0.35,1, 0.7,2, 1,3:
      8,0.35,1, 0.65,2, 1,3:
      9,0.30,1, 0.65,2, 1,3:
      10,180:
      11,360:
      12,600:
      13,20:
      14,20:
```

Appendix 25 continued

```
     15,20:
     16,30:
     17,30:
     18,30:
     19,30:
     20,30:
     21,30:
     22,180,30:
     23,360,60:
     24,600,100:
     25,2:
     26,2.5:
     27,5:
     28,21600:
     29,14400:
     30,7200:
     31,10:
     32,25:
     33,80:
     34,4320:
     35,7200:
     36,14400:
     37,200,50:
     38,360,80:
     39,420,120:
     40,10000:
     41,25000:
     42,60000:
     43,100:
     44,60:
     45,30:
     46,3000:
     47,7500:
     48,24000:
     49,4320,420:
     50,5760,600:
     51,7200,720:
     52,30:
     53,20:
     54,20:
     55,20:
     56,150:
     57,1:
     58,1:
     59,1;
TALLIES:WAITING_TIME_FOR_DRG1:
     WAITING_TIME_FOR_DRG2:
     WAITING_TIME_FOR_DRG3:
     AVERAGE_DRG1_WAITING_TIME:
     AVERAGE_DRG2_WAITING_TIME:
```

Appendix 25 continued

```
     AVERAGE_DRG3_WAITING_TIME;
COUNTERS:GARBAGE:
     SURGERY_OF_DRG1:
     SURGERY_OF_DRG2:
     SURGERY_OF_DRG3:
     TOTAL_DRG1:
     TOTAL_DRG2:
     TOTAL_DRG3:
     NO_USE:
     TOTAL_DELAYED_COST:
     SPE_EMG:
     TOTAL_WAIT_COST;
DSTATS:NR(DOCTOR(1)):
     NR(DOCTOR(2)):
     NR(DOCTOR(3)):
     NR(OPERATING_ROOM):
     NR(EMG_BED):
     NR(BED):
     NQ(9):
     NQ(10):
     NQ(11):
     NQ(12):
     NQ(13):
     NQ(14):
     NQ(15):
     NQ(16):
     NQ(17):
     NQ(18):
     NQ(19):
     NQ(20):
     NQ(23);
REPLICATE,1,0,784800,,,655200;
END;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

Appendix 25 continued

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT A1 Run execution date : 11/26/1994
Analyst: WEN_YEN CHEN Model revision date: 11/26/1994

Replication ended at time : 784800.
Statistics were cleared at time: 655200.
Statistics accumulated for time: 129600.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	6528.4	1.0640	.00000	31464.	89
WAITING_TIME_FOR_DRG2	2226.5	1.1729	.00000	12600.	64
WAITING_TIME_FOR_DRG3	863.15	1.1380	.00000	3774.1	21
AVERAGE_DRG1_WAITING_T	2.8723	.41227	2.0000	8.5831	89
AVERAGE_DRG2_WAITING_T	2.9732	.22825	2.5000	5.9972	64
AVERAGE_DRG3_WAITING_T	5.6906	.15170	5.0000	8.4453	21

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.25436	1.7122	.00000	1.0000	.00000
NR(DOCTOR(2))	.27173	1.6371	.00000	1.0000	.00000
NR(DOCTOR(3))	.21648	1.9025	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.39845	1.2287	.00000	1.0000	.00000
NR(EMG_BED)	.32263	1.6297	.00000	2.0000	.00000
NR(BED)	14.306	.25118	5.0000	22.000	6.0000
NQ(12)	.03751	5.0656	.00000	1.0000	.00000
NQ(13)	.04097	4.8385	.00000	1.0000	.00000
NQ(14)	.02266	6.5672	.00000	1.0000	.00000
NQ(15)	.01140	9.3115	.00000	1.0000	.00000
NQ(16)	.02688	6.0168	.00000	1.0000	.00000
NQ(17)	.02888	5.7987	.00000	1.0000	.00000
NQ(18)	.01922	7.1443	.00000	1.0000	.00000
NQ(19)	.01082	9.5622	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(22)	5.0073	.53435	.00000	11.000	.00000
NQ(28)	5.2532	1.0795	.00000	31.000	10.000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	92	Infinite
SURGERY_OF_DRG2	66	Infinite
SURGERY_OF_DRG3	21	Infinite
TOTAL_DRG1	86	Infinite
TOTAL_DRG2	66	Infinite
TOTAL_DRG3	23	Infinite
NO_USE	0	Infinite
TOTAL_DELAYED_COST	565	Infinite
SPE_EMG	1	Infinite
TOTAL_WAIT_COST	515	Infinite

Run Time: 1 min(s) 57 sec(s)
Simulation run complete.

Appendix 26 output file of model A1

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT A2 | Run execution date : 11/27/1994
Analyst: WEN_YEN CHEN | Model revision date: 11/27/1994

Replication ended at time : 784800.
Statistics were cleared at time: 655200.
Statistics accumulated for time: 129600.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	10044.	1.1671	.00000	61251.	89
WAITING_TIME_FOR_DRG2	1806.3	1.0308	.00000	11420.	59
WAITING_TIME_FOR_DRG3	1159.0	.91602	45.688	3960.0	24
AVERAGE_DRG1_WAITING_T	3.9866	1.1264	2.0000	34.085	89
AVERAGE_DRG2_WAITING_T	2.8603	.15549	2.5000	5.5253	59
AVERAGE_DRG3_WAITING_T	5.9377	.15849	5.0318	8.6663	24

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.23335	1.8126	.00000	1.0000	.00000
NR(DOCTOR(2))	.24213	1.7692	.00000	1.0000	.00000
NR(DOCTOR(3))	.24159	1.7718	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.39983	1.2252	.00000	1.0000	.00000
NR(EMG_BED)	.75919	1.1275	.00000	3.0000	1.0000
NR(BED)	15.375	.33539	5.0000	29.000	13.000
NQ(4)	.00000	--	.00000	.00000	.00000
NQ(12)	.02997	5.6888	.00000	1.0000	.00000
NQ(13)	.03426	5.3091	.00000	1.0000	.00000
NQ(14)	.03640	5.1453	.00000	1.0000	.00000
NQ(15)	.00915	10.407	.00000	1.0000	.00000
NQ(16)	.02225	6.6286	.00000	1.0000	.00000
NQ(17)	.02455	6.3032	.00000	1.0000	.00000
NQ(18)	.03127	5.5656	.00000	1.0000	.00000
NQ(19)	.00821	10.992	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(22)	6.0021	.69018	.00000	16.000	2.0000
NQ(28)	4.8664	1.1341	.00000	32.000	19.000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	90	Infinite
SURGERY_OF_DRG2	55	Infinite
SURGERY_OF_DRG3	26	Infinite
TOTAL_DRG1	83	Infinite
TOTAL_DRG2	65	Infinite
TOTAL_DRG3	30	Infinite
NO_USE	1	Infinite
TOTAL_DELAYED_COST	666	Infinite
SPE_EMG	1	Infinite
TOTAL_WAIT_COST	714	Infinite

Run Time: 2 min(s) 42 sec(s)
Simulation run complete.

Appendix 27 output file of model A2

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT A3 Run execution date : 11/27/1994
Analyst: WEN_YEN CHEN Model revision date: 11/27/1994

Replication ended at time : 784800.
Statistics were cleared at time: 655200.
Statistics accumulated for time: 129600.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	30896.	.91974	.00000	.12951E+06	97
WAITING_TIME_FOR_DRG2	2430.8	1.0629	.00000	10020.	54
WAITING_TIME_FOR_DRG3	1246.4	.86385	.00000	3962.1	33
AVERAGE_DRG1_WAITING_T	32.378	3.2079	2.0000	803.50	97
AVERAGE_DRG2_WAITING_T	3.0107	.20283	2.5000	5.0134	54
AVERAGE_DRG3_WAITING_T	6.0126	.15961	5.0000	8.6687	33

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.26934	1.6471	.00000	1.0000	.00000
NR(DOCTOR(2))	.26786	1.6533	.00000	1.0000	.00000
NR(DOCTOR(3))	.23435	1.8075	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.44151	1.1247	.00000	1.0000	1.0000
NR(EMG_BED)	.60350	1.1304	.00000	3.0000	.00000
NR(BED)	30.193	.19537	14.000	36.000	14.000
NQ(12)	.05007	4.3555	.00000	1.0000	.00000
NQ(13)	.04079	4.8490	.00000	1.0000	.00000
NQ(14)	.03561	5.2044	.00000	1.0000	.00000
NQ(15)	.01444	8.2629	.00000	1.0000	.00000
NQ(16)	.02291	6.5300	.00000	1.0000	.00000
NQ(17)	.03439	5.2988	.00000	1.0000	.00000
NQ(18)	.02033	6.9421	.00000	1.0000	.00000
NQ(19)	.00858	10.750	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(22)	19.957	.29977	1.0000	30.000	4.0000
NQ(28)	6.5323	1.0379	.00000	33.000	19.000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	95	Infinite
SURGERY_OF_DRG2	56	Infinite
SURGERY_OF_DRG3	31	Infinite
TOTAL_DRG1	80	Infinite
TOTAL_DRG2	60	Infinite
TOTAL_DRG3	36	Infinite
NO_USE	0	Infinite
TOTAL_DELAYED_COST	3501	Infinite
SPE_EMG	2	Infinite
TOTAL_WAIT_COST	2200	Infinite

Run Time: 3 min(s) 19 sec(s)
Simulation run complete.

Appendix 28 output file of model A3

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT B1 Run execution date : 11/26/1994
Analyst: WEN_YEN CHEN Model revision date: 11/26/1994

Replication ended at time : 784800.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	1896.0	1.0187	.00000	9990.0	411
WAITING_TIME_FOR_DRG2	1954.6	1.0030	.00000	9980.0	319
WAITING_TIME_FOR_DRG3	1772.4	1.1938	.00000	11490.	136
AVERAGE_DRG1_WAITING_T	2.1927	.09660	2.0000	3.1761	411
AVERAGE_DRG2_WAITING_T	2.8920	.15204	2.5000	4.9995	319
AVERAGE_DRG3_WAITING_T	6.7587	.44594	5.0000	24.662	136

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.22018	1.8820	.00000	1.0000	.00000
NR(DOCTOR(2))	.22251	1.8693	.00000	1.0000	.00000
NR(DOCTOR(3))	.20886	1.9462	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.34787	1.3692	.00000	1.0000	.00000
NR(EMG_BED)	.49814	1.5309	.00000	4.0000	.00000
NR(BED)	10.063	.35011	.00000	21.000	6.0000
NQ(12)	.02694	6.0101	.00000	1.0000	.00000
NQ(13)	.03442	5.2965	.00000	1.0000	.00000
NQ(14)	.03256	5.4511	.00000	1.0000	.00000
NQ(15)	.01227	8.9730	.00000	1.0000	.00000
NQ(16)	.02194	6.6767	.00000	1.0000	.00000
NQ(17)	.02527	6.2112	.00000	1.0000	.00000
NQ(18)	.02507	6.2359	.00000	1.0000	.00000
NQ(19)	.00715	11.786	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(22)	2.0893	.92801	.00000	10.000	.00000
NQ(28)	3.5460	1.1845	.00000	33.000	14.000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	411	Infinite
SURGERY_OF_DRG2	317	Infinite
SURGERY_OF_DRG3	132	Infinite
TOTAL_DRG1	85	Infinite
TOTAL_DRG2	68	Infinite
TOTAL_DRG3	26	Infinite
NO_USE	1	Infinite
TOTAL_DELAYED_COST	529	Infinite
SPE_EMG	4	Infinite
TOTAL_WAIT_COST	263	Infinite

Run Time: 2 min(s) 33 sec(s)
Simulation run complete.

Appendix 29 output file of model B1

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT B2 Run execution date : 11/27/1994
Analyst: WEN_YEN CHEN Model revision date: 11/27/1994

Replication ended at time : 784800.
Statistics were cleared at time: 655200.
Statistics accumulated for time: 129600.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	3887.1	1.3458	.00000	25740.	83
WAITING_TIME_FOR_DRG2	4476.2	1.0857	.00000	19789.	63
WAITING_TIME_FOR_DRG3	2441.0	.89554	.00000	8935.1	24
AVERAGE_DRG1_WAITING_T	2.4826	.34258	2.0000	6.5851	83
AVERAGE_DRG2_WAITING_T	3.6406	.42680	2.5000	9.8801	63
AVERAGE_DRG3_WAITING_T	7.3748	.36978	5.0000	17.295	24

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.24794	1.7416	.00000	1.0000	.00000
NR(DOCTOR(2))	.24340	1.7631	.00000	1.0000	.00000
NR(DOCTOR(3))	.24246	1.7676	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.40888	1.2024	.00000	1.0000	.00000
NR(EMG_BED)	.50402	1.0858	.00000	2.0000	1.0000
NR(BED)	13.723	.30372	6.0000	24.000	10.000
NQ(4)	.00000	--	.00000	.00000	.00000
NQ(12)	.03306	5.4078	.00000	1.0000	.00000
NQ(13)	.03104	5.5872	.00000	1.0000	.00000
NQ(14)	.03987	4.9074	.00000	1.0000	.00000
NQ(15)	.02183	6.6940	.00000	1.0000	.00000
NQ(16)	.02116	6.8013	.00000	1.0000	.00000
NQ(17)	.02612	6.1062	.00000	1.0000	.00000
NQ(18)	.02441	6.3218	.00000	1.0000	.00000
NQ(19)	.01386	8.4339	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(22)	4.2615	.83125	.00000	12.000	1.0000
NQ(28)	5.3442	1.2349	.00000	35.000	18.000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	81	Infinite
SURGERY_OF_DRG2	62	Infinite
SURGERY_OF_DRG3	27	Infinite
TOTAL_DRG1	83	Infinite
TOTAL_DRG2	64	Infinite
TOTAL_DRG3	28	Infinite
NO_USE	0	Infinite
TOTAL_DELAYED_COST	612	Infinite
SPE_EMG	1	Infinite
TOTAL_WAIT_COST	460	Infinite

Run Time: 3 min(s) 34 sec(s)
Simulation run complete.

Appendix 30 output file of model B2

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT B3 Run execution date : 11/27/1994
Analyst: WEN_YEN CHEN Model revision date: 11/27/1994

Replication ended at time : 784800.
Statistics were cleared at time: 655200.
Statistics accumulated for time: 129600.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	7722.2	.93430	.00000	33110.	87
WAITING_TIME_FOR_DRG2	8079.5	.93810	.00000	32729.	58
WAITING_TIME_FOR_DRG3	5680.2	1.0138	.00000	21871.	33
AVERAGE_DRG1_WAITING_T	3.0540	.44743	2.0000	9.2629	87
AVERAGE_DRG2_WAITING_T	5.2388	.82857	2.5000	24.267	58
AVERAGE_DRG3_WAITING_T	16.979	1.3369	5.0000	104.28	33

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.25986	1.6877	.00000	1.0000	.00000
NR(DOCTOR(2))	.23341	1.8123	.00000	1.0000	.00000
NR(DOCTOR(3))	.27641	1.6180	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.44413	1.1187	.00000	1.0000	1.0000
NR(EMG_BED)	.50306	1.3725	.00000	2.0000	1.0000
NR(BED)	18.805	.28080	7.0000	30.000	7.0000
NQ(12)	.04062	4.8598	.00000	1.0000	.00000
NQ(13)	.03621	5.1594	.00000	1.0000	.00000
NQ(14)	.05851	4.0113	.00000	1.0000	.00000
NQ(15)	.01987	7.0227	.00000	1.0000	.00000
NQ(16)	.03669	5.1237	.00000	1.0000	.00000
NQ(17)	.01611	7.8156	.00000	1.0000	.00000
NQ(18)	.02429	6.3380	.00000	1.0000	.00000
NQ(19)	.02154	6.7392	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(22)	8.5247	.54476	.00000	19.000	.00000
NQ(28)	6.0025	1.0791	.00000	27.000	5.0000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	89	Infinite
SURGERY_OF_DRG2	60	Infinite
SURGERY_OF_DRG3	32	Infinite
TOTAL_DRG1	84	Infinite
TOTAL_DRG2	57	Infinite
TOTAL_DRG3	34	Infinite
NO_USE	1	Infinite
TOTAL_DELAYED_COST	1129	Infinite
SPE_EMG	1	Infinite
TOTAL_WAIT_COST	922	Infinite

Run Time: 3 min(s) 37 sec(s)
Simulation run complete.

Appendix 31 output file of model B3

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT C1 Run execution date : 11/26/1994
Analyst: WEN_YEN CHEN Model revision date: 11/26/1994

Replication ended at time : 784800.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	2745.6	1.5639	.00000	38918.	411
WAITING_TIME_FOR_DRG2	1614.1	1.3249	.00000	15740.	319
WAITING_TIME_FOR_DRG3	795.88	1.0928	.00000	4426.5	136
AVERAGE_DRG1_WAITING_T	2.3297	.31353	2.0000	12.120	411
AVERAGE_DRG2_WAITING_T	2.8325	.18997	2.5000	7.4584	319
AVERAGE_DRG3_WAITING_T	5.6286	.13785	5.0000	9.2463	136

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.25497	1.7094	.00000	1.0000	.00000
NR(DOCTOR(2))	.20429	1.9736	.00000	1.0000	.00000
NR(DOCTOR(3))	.19786	2.0135	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.34904	1.3656	.00000	1.0000	1.0000
NR(EMG_BED)	.44991	1.5497	.00000	4.0000	.00000
NR(BED)	10.217	.39047	.00000	24.000	7.0000
NQ(9)	.62046	1.5289	.00000	8.0000	.00000
NQ(10)	.81299	1.3345	.00000	5.0000	.00000
NQ(11)	.79488	1.2514	.00000	6.0000	.00000
NQ(12)	.02971	5.7147	.00000	1.0000	.00000
NQ(13)	.02950	5.7355	.00000	1.0000	.00000
NQ(14)	.02833	5.8568	.00000	1.0000	.00000
NQ(15)	.02194	6.6764	.00000	1.0000	.00000
NQ(16)	.02343	6.4563	.00000	1.0000	.00000
NQ(17)	.01869	7.2451	.00000	1.0000	.00000
NQ(18)	.02041	6.9282	.00000	1.0000	.00000
NQ(19)	.00979	10.055	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(23)	4.7501	1.1747	.00000	33.000	6.0000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	410	Infinite
SURGERY_OF_DRG2	317	Infinite
SURGERY_OF_DRG3	133	Infinite
TOTAL_DRG1	83	Infinite
TOTAL_DRG2	68	Infinite
TOTAL_DRG3	22	Infinite
NO_USE	1	Infinite
TOTAL_DELAYED_COST	519	Infinite
SPE_EMG	4	Infinite
TOTAL_WAIT_COST	342	Infinite

Run Time: 2 min(s) 48 sec(s)
Simulation run complete.

Appendix 32 output file of model C1

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT C2 Run execution date : 11/28/1994
Analyst: WEN_YEN CHEN Model revision date: 11/28/1994

Replication ended at time : 784800.
Statistics were cleared at time: 655200.
Statistics accumulated for time: 129600.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	3228.6	1.5794	.00000	28574.	81
WAITING_TIME_FOR_DRG2	1664.2	1.2463	.00000	12790.	60
WAITING_TIME_FOR_DRG3	538.51	1.2378	.00000	2770.0	24
AVERAGE_DRG1_WAITING_T	2.4038	.33811	2.0000	7.5085	81
AVERAGE_DRG2_WAITING_T	2.8397	.18363	2.5000	6.0768	60
AVERAGE_DRG3_WAITING_T	5.4116	.10021	5.0000	7.3460	24

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.25773	1.6971	.00000	1.0000	.00000
NR(DOCTOR(2))	.24808	1.7410	.00000	1.0000	.00000
NR(DOCTOR(3))	.21182	1.9290	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.39447	1.2390	.00000	1.0000	1.0000
NR(EMG_BED)	.33395	1.5454	.00000	2.0000	.00000
NR(BED)	11.654	.33063	4.0000	23.000	9.0000
NQ(4)	.00000	--	.00000	.00000	.00000
NQ(9)	.71841	1.2734	.00000	3.0000	.00000
NQ(10)	.81995	1.2622	.00000	4.0000	.00000
NQ(11)	.95705	1.2615	.00000	7.0000	.00000
NQ(12)	.04220	4.7641	.00000	1.0000	.00000
NQ(13)	.02130	6.7785	.00000	1.0000	.00000
NQ(14)	.02764	5.9311	.00000	1.0000	.00000
NQ(15)	.02084	6.8549	.00000	1.0000	.00000
NQ(16)	.03328	5.3895	.00000	1.0000	.00000
NQ(17)	.03886	4.9733	.00000	1.0000	.00000
NQ(18)	.01909	7.1678	.00000	1.0000	.00000
NQ(19)	.00823	10.976	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(23)	4.5787	1.2066	.00000	29.000	12.000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	83	Infinite
SURGERY_OF_DRG2	59	Infinite
SURGERY_OF_DRG3	25	Infinite
TOTAL_DRG1	82	Infinite
TOTAL_DRG2	63	Infinite
TOTAL_DRG3	24	Infinite
NO_USE	0	Infinite
TOTAL_DELAYED_COST	494	Infinite
SPE_EMG	1	Infinite
TOTAL_WAIT_COST	259	Infinite

Run Time: 2 min(s) 18 sec(s)
Simulation run complete.

Appendix 33 output file of model C2

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT C3 Run execution date : 11/28/1994
Analyst: WEN_YEN CHEN Model revision date: 11/28/1994

Replication ended at time : 784800.
Statistics were cleared at time: 655200.
Statistics accumulated for time: 129600.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	11589.	1.5130	.00000	76319.	91
WAITING_TIME_FOR_DRG2	2609.5	1.1001	.00000	12490.	55
WAITING_TIME_FOR_DRG3	1686.8	1.3618	.00000	9719.0	31
AVERAGE_DRG1_WAITING_T	5.9212	1.8169	2.0000	68.472	91
AVERAGE_DRG2_WAITING_T	3.0617	.23201	2.5000	5.9516	55
AVERAGE_DRG3_WAITING_T	6.7333	.46890	5.0000	19.284	31

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.25786	1.6965	.00000	1.0000	.00000
NR(DOCTOR(2))	.24325	1.7638	.00000	1.0000	.00000
NR(DOCTOR(3))	.25490	1.7097	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.43088	1.1493	.00000	1.0000	1.0000
NR(EMG_BED)	.24988	1.9418	.00000	2.0000	.00000
NR(BED)	15.841	.34697	3.0000	27.000	9.0000
NQ(9)	1.8923	.70996	.00000	6.0000	.00000
NQ(10)	1.9559	.76095	.00000	5.0000	.00000
NQ(11)	2.0644	.72684	.00000	6.0000	.00000
NQ(12)	.03275	5.4349	.00000	1.0000	.00000
NQ(13)	.03739	5.0737	.00000	1.0000	.00000
NQ(14)	.03820	5.0175	.00000	1.0000	.00000
NQ(15)	.03009	5.6775	.00000	1.0000	.00000
NQ(16)	.02044	6.9224	.00000	1.0000	.00000
NQ(17)	.02605	6.1142	.00000	1.0000	.00000
NQ(18)	.02827	5.8630	.00000	1.0000	.00000
NQ(19)	.01839	7.3054	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(23)	6.0135	1.1054	.00000	39.000	13.000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	96	Infinite
SURGERY_OF_DRG2	54	Infinite
SURGERY_OF_DRG3	30	Infinite
TOTAL_DRG1	79	Infinite
TOTAL_DRG2	58	Infinite
TOTAL_DRG3	32	Infinite
NO_USE	0	Infinite
TOTAL_DELAYED_COST	915	Infinite
SPE_EMG	0	Infinite
TOTAL_WAIT_COST	868	Infinite

Run Time: 2 min(s) 25 sec(s)
Simulation run complete.

Appendix 34 output file of model C3

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT D1 Run execution date : 11/26/1994
Analyst: WEN_YEN CHEN Model revision date: 11/26/1994

Replication ended at time : 784800.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	2383.1	1.2353	.00000	19709.	412
WAITING_TIME_FOR_DRG2	1350.3	1.3125	.00000	12489.	319
WAITING_TIME_FOR_DRG3	646.45	.97239	.00000	2850.0	136
AVERAGE_DRG1_WAITING_T	2.2563	.16046	2.0000	4.9809	412
AVERAGE_DRG2_WAITING_T	2.7690	.14701	2.5000	5.9511	319
AVERAGE_DRG3_WAITING_T	5.4910	.09117	5.0000	7.4281	136

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.25025	1.7309	.00000	1.0000	.00000
NR(DOCTOR(2))	.21202	1.9278	.00000	1.0000	.00000
NR(DOCTOR(3))	.19584	2.0264	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.34714	1.3714	.00000	1.0000	.00000
NR(EMG_BED)	.43699	1.5422	.00000	4.0000	.00000
NR(BED)	9.8928	.36294	.00000	22.000	7.0000
NQ(9)	.57025	1.4992	.00000	4.0000	.00000
NQ(10)	.68702	1.4205	.00000	5.0000	.00000
NQ(11)	.64960	1.3371	.00000	4.0000	.00000
NQ(12)	.02565	6.1633	.00000	1.0000	.00000
NQ(13)	.03286	5.4253	.00000	1.0000	.00000
NQ(14)	.02706	5.9959	.00000	1.0000	.00000
NQ(15)	.01758	7.4760	.00000	1.0000	.00000
NQ(16)	.02479	6.2720	.00000	1.0000	.00000
NQ(17)	.02049	6.9133	.00000	1.0000	.00000
NQ(18)	.02187	6.6883	.00000	1.0000	.00000
NQ(19)	.00957	10.171	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(23)	4.3274	1.2203	.00000	35.000	11.000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	410	Infinite
SURGERY_OF_DRG2	317	Infinite
SURGERY_OF_DRG3	133	Infinite
TOTAL_DRG1	88	Infinite
TOTAL_DRG2	64	Infinite
TOTAL_DRG3	25	Infinite
NO_USE	1	Infinite
TOTAL_DELAYED_COST	509	Infinite
SPE_EMG	4	Infinite
TOTAL_WAIT_COST	321	Infinite

Run Time: 2 min(s) 48 sec(s)
Simulation run complete.

Appendix 35 output file of model D1

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT D2 Run execution date : 11/27/1994
Analyst: WEN_YEN CHEN Model revision date: 11/27/1994

Replication ended at time : 784800.
Statistics were cleared at time: 655200.
Statistics accumulated for time: 129600.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	2699.3	1.3813	.00000	20040.	80
WAITING_TIME_FOR_DRG2	1280.2	1.3311	.00000	8428.6	57
WAITING_TIME_FOR_DRG3	436.68	1.1772	.00000	1428.9	24
AVERAGE_DRG1_WAITING_T	2.3065	.22898	2.0000	5.0578	80
AVERAGE_DRG2_WAITING_T	2.7532	.13770	2.5000	4.4889	57
AVERAGE_DRG3_WAITING_T	5.3258	.07321	5.0000	6.0976	24

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.26025	1.6860	.00000	1.0000	.00000
NR(DOCTOR(2))	.23616	1.7985	.00000	1.0000	.00000
NR(DOCTOR(3))	.20327	1.9798	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.38632	1.2604	.00000	1.0000	1.0000
NR(EMG_BED)	.55084	1.3242	.00000	3.0000	1.0000
NR(BED)	11.046	.27667	4.0000	18.000	10.000
NQ(4)	.00000	--	.00000	.00000	.00000
NQ(9)	.50134	1.4883	.00000	4.0000	.00000
NQ(10)	.73628	1.2048	.00000	3.0000	.00000
NQ(11)	.82482	1.1766	.00000	4.0000	1.0000
NQ(12)	.03848	4.9989	.00000	1.0000	.00000
NQ(13)	.03835	5.0077	.00000	1.0000	.00000
NQ(14)	.03217	5.4848	.00000	1.0000	.00000
NQ(15)	.01766	7.4575	.00000	1.0000	.00000
NQ(16)	.01944	7.1023	.00000	1.0000	.00000
NQ(17)	.02999	5.6873	.00000	1.0000	.00000
NQ(18)	.02270	6.5620	.00000	1.0000	.00000
NQ(19)	.01185	9.1308	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(23)	3.8142	1.1239	.00000	28.000	10.000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	81	Infinite
SURGERY_OF_DRG2	58	Infinite
SURGERY_OF_DRG3	25	Infinite
TOTAL_DRG1	82	Infinite
TOTAL_DRG2	62	Infinite
TOTAL_DRG3	29	Infinite
NO_USE	0	Infinite
TOTAL_DELAYED_COST	469	Infinite
SPE_EMG	0	Infinite
TOTAL_WAIT_COST	207	Infinite

Run Time: 3 min(s) 37 sec(s)
Simulation run complete.

Appendix 36 output file of model D2

SIMAN IV - License #9210474
Lehigh University

Summary for Replication 1 of 1

Project: INPATIENT D3
Analyst: WEN_YEN CHEN

Run execution date : 11/27/1994
Model revision date: 11/27/1994

Replication ended at time : 784800.
Statistics were cleared at time: 655200.
Statistics accumulated for time: 129600.

TALLY VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Observations
WAITING_TIME_FOR_DRG1	8152.9	1.0341	.00000	32031.	91
WAITING_TIME_FOR_DRG2	4570.4	1.1461	.00000	18660.	55
WAITING_TIME_FOR_DRG3	1503.9	1.1090	.00000	6800.0	31
AVERAGE_DRG1_WAITING_T	3.1751	.47286	2.0000	8.8114	91
AVERAGE_DRG2_WAITING_T	3.7012	.45356	2.5000	9.1351	55
AVERAGE_DRG3_WAITING_T	6.3459	.28289	5.0000	12.857	31

DISCRETE-CHANGE VARIABLES

Identifier	Average	Variation	Minimum	Maximum	Final Value
NR(DOCTOR(1))	.28429	1.5867	.00000	1.0000	.00000
NR(DOCTOR(2))	.25211	1.7223	.00000	1.0000	.00000
NR(DOCTOR(3))	.23438	1.8074	.00000	1.0000	.00000
NR(OPERATING_ROOM)	.43731	1.1343	.00000	1.0000	.00000
NR(EMG_BED)	.65634	1.0786	.00000	3.0000	.00000
NR(BED)	16.366	.33456	4.0000	27.000	8.0000
NQ(9)	1.9853	.78081	.00000	6.0000	.00000
NQ(10)	2.3365	.70944	.00000	6.0000	.00000
NQ(11)	2.0295	.67286	.00000	6.0000	.00000
NQ(12)	.04152	4.8047	.00000	1.0000	.00000
NQ(13)	.03179	5.5191	.00000	1.0000	.00000
NQ(14)	.04515	4.5987	.00000	1.0000	.00000
NQ(15)	.03475	5.2706	.00000	1.0000	.00000
NQ(16)	.02802	5.8896	.00000	1.0000	.00000
NQ(17)	.01882	7.2211	.00000	1.0000	.00000
NQ(18)	.01940	7.1090	.00000	1.0000	.00000
NQ(19)	.02124	6.7889	.00000	1.0000	.00000
NQ(20)	.00000	--	.00000	.00000	.00000
NQ(23)	6.5280	1.0516	.00000	30.000	9.0000

COUNTERS

Identifier	Count	Limit
GARBAGE	0	Infinite
SURGERY_OF_DRG1	92	Infinite
SURGERY_OF_DRG2	58	Infinite
SURGERY_OF_DRG3	30	Infinite
TOTAL_DRG1	82	Infinite
TOTAL_DRG2	60	Infinite
TOTAL_DRG3	36	Infinite
NO_USE	0	Infinite
TOTAL_DELAYED_COST	689	Infinite
SPE_EMG	1	Infinite
TOTAL_WAIT_COST	722	Infinite

Run Time: 3 min(s) 31 sec(s)
Simulation run complete.

Appendix 37 output file of model D3

References:

(1) Banker, R.D., Conrad, R.F., and Strauss, R.P.(1986), "A Comparative Application of Data Envelopment Analysis and Translog Methods: An Illustrative Study of Hospital Production", Management Science, Vol.32, No.1, 30-44.

(2) Buzacott, J.A. and Shanthikumar (1980), "Models for Understanding Flexible Manufacturing Systems", AIIE, Vol. 12, No.4, 339-350.

(3) Charnes, A., and Cooper, W.W., and Rhodes, E.(1978), "Measuring the Efficiency of Decision Making Unit", European Journal of Operations Research, 429-444.

(4) Cohen, A.C.(1991), Truncated and Censored Samples: Theory and Applications, Marcel Dekker, Inc.

(5) Conway, T.W. (1965 a), "Priority Dispatching and Work in Process Inventory in a Job Shop", Journal of Industrial Engineering 16, 123-130.

(6) Conway, T.W. (1965 b), "Priority Dispatching and Job Lateness in a Job Shop", Journal of Industrial Engineering 11, 221-229.

(7) Congressional Information Service (1993), "Statistical Reference Index: A selective guide to American Statistical Publications from resources other than the U.S. government", Index No. B4455-1~3.

(8) Freidenfelds, J. (1981), "Capacity Expansion: Analysis of Simple Models with Applications", Elsevier North Holland

Inc.

(9) Hax, A.C. and Candea, D. (1984), "Production and Inventory Management", Prentice-Hall Inc.

(10) Hershey, J.C., Weiss, E.N., and Cohen, M.A.(1981), "A Stochastic Service Network Model with Application to Hospital Facilities", Operations Research, Vol.29, No.1, 1-22.

(11) Hornbrook, M.C. (1983), "Allocative Mechine:Efficiency, Disease Severity, and the Payment Mechanism", Annal of the American Academy of Political and Social Science 468, 17.

(12) Howard, R.A. (1971), Dynamic Probabilistic Systems, Vol.2: Semi-Markov and Decision Processes, John Wiley and Sons, Inc.

(13) Hutchison, K.L., Snyder, D., and Ward, F.(1989), "Scheduling for Random Job Shop Flexible Manufacturing Systems", Proceedings of the Third ORSA/TIMS Conference on Flexible Manufacturing Systems: Operations Research Models and Applications, 161-166.

(14) Johnson, L.A. and Montgomery, D.C. (1974), Operations Research in Production Planning, Scheduling, and Inventory Control. John Wiley & Sons, Inc.

(15) Kao, E.P.C.(1973), "A Semi-Markovian Population Model with Application to Hospital Planning", IEEE Transactions on System, Man, and Cybernetics, Vol.SMC-3, No.4, 327-336.

(16) Kao, E.P.C.(1974), "Modeling the Movement of Coronary

Patients within a Hospital by Semi-Markov Processes", Operations Research, Vol.22, No.4, 683-699.

(17) Kao, E.P.C.(1974), "A Note on the First Two Moments of Times in Transient States in a Semi-Markov Process", Journal of Applied Probability, Vol.11, 193-198.

(18) Kao, E.P.C., and Queyranne, M.(1985), "Budgeting Costs of Nursing in a Hospital", Management Science, Vol.31, No.5, 608-621.

(19) Mirchandani, P.B., and Lee, E.J.(1988), "Concurrent Routing, Sequencing, and Setups for a Two-Machine Flexible Manufacturing Cell", IEEE Journal of Robotics and Automation, Vol.4, No.3, 256-264.

(20) Morey, R., and Dittman, D.A.(1984), "Hospital Profit Planning under Medicare Reimbursement", Operations Research, Vol.32, No.2, 250-269.

(21) Morton, T.E., and Pentico, D.W. (1993), Heuristic Scheduling Systems : with Applications to Production Systems and Project Management, John Wiley and Sons, Inc.

(22) Mukhopadhyay, S.K., Maiti, B., and Garg, S.(1991), "Heuristic Solution to the Scheduling Problems in Flexible Manufacturing System", International Journal of Production Research, Vol.29, No.10, 2003-2024.

(23) Polesel, P., and Romanin-Jacur, G.(1986), "A Application of Semi-Markov Processes to Health Care Planning: a model to study the flow of patients following the dialysis-transplantation plan", Semi-Markov Models:

Theory and Applications, Edited by Janssen, J. Plenum Press.

(24) Shanker, K., and Tzen, Y.J.(1985), "A Loading and Dispatching Problem in a Random Flexible Manufacturing System", International Journal of Production Research, Vol. 23, No.3, 579-595.

(25) Stecke, K.E., and Solberg, J.J.(1981), "Loading and Control Policies for a Flexible Manufacturing System", International Journal of Production Research, Vol.19, No.5, 481-490.

(26) Stecke, K.E. (1983),, "Formulation and Solution of Nonlinear Integer Production Planning for Flexible Manufacturing Systems", Management Science, Vol.29, 273-288.

(27) Tenenbaum, A., and Seidman, A.(1989), "Dynamic Load Control Policies for a Flexible Manufacturing System with Stochastic Processing Rates", International Journal of Flexible Manufacturing Systems. Vol.2, 93-120.

(28) Vincent, K.O. (1991), Total Quality and Productivity Management in Health Care Organizations, American Society for Quality Control, Industrial Engineering and Management Press, Institute of Industrial Engineers.

(29) Williams, S.J., and Torrens, P.R.(1993), Introduction to Health Services, 4th edition, Delmar Publishers Inc.

(30) Winston, W.L. (1994), "Operations Research: Applications and Algorithms", third edition, International Thomson Publishing.

(31) Woodbury, M.A., and Manton, K.G.(1993), "A Model for

Allocating Budgets in a Closed System which Simultaneously Computes DRG Allocation Weight", Operations Research, vol.41, No.2, 298-309.

Vita

The author was born on August 25th, 1967 in Taipei, Taiwan. He received a Bachelor of Mechanical Engineering Degree from N.C.T.U. in HsinChu, Taiwan in 1989. During the period 1989 to 1991, he serviced as a tank platoon commander in Taiwan Army.

END
OF
TITLE